BATCH-NORM STATISTICS: THE UNRECORDED CONVENTION IN UNLEARNING AUDITS

Anonymous authors
Paper under double-blind review

ABSTRACT

Unlearning benchmarks release checkpoints with tables of accuracies and pass/fail verdicts, and a reader treats each entry as a property of the released checkpoint. A batch-normalized checkpoint, however, also ships normalization statistics fitted from data. An audit's retrained reference fits them on retained data by construction, while the unlearned checkpoint ships whatever its own run left, and no release we audit records how. We show that the published numbers of such checkpoints, and some verdicts, also depend on this fitting convention. We reproduce each checkpoint's published numbers and then refit only its statistics on retained data at byte-identical weights. The refit raises forget accuracy beyond the spread MU-Bench reports across training seeds on 47 of the 221 reproducible MU-Bench checkpoints we test. A method's average does not predict which ones. In a targeted test, removed data in the statistics does not explain which checkpoints move. Releases should therefore state how their statistics were fitted, and a reader can check one checkpoint on a single GPU in under an hour.

1 INTRODUCTION

Machine unlearning removes the influence of a subset of the training data from an already trained model (Cao & Yang, 2015; Bourtoule et al., 2021; Guo et al., 2020). An audit decides whether a removal worked by comparing the unlearned model with a reference retrained from scratch without that subset. It reads three accuracies: on the removed data (forget accuracy d_f), on the kept training data (retain accuracy d_r), and on held-out test data (test accuracy d_t) (Golatkar et al., 2020; Triantafillou et al., 2024). Benchmarks now release unlearned checkpoints together with a table of these accuracies and a pass/fail criterion computed from them (Cheng & Amiri, 2024; Lee et al., 2026). A reader who cites such a table treats each number, and each verdict, as a property of the released checkpoint.

Consider one checkpoint that MU-Bench releases, a CIFAR-10 model unlearned with the bad-teaching method (Chundawat et al., 2023). With the statistics it ships, it reproduces its published numbers, and its criterion value, the gap $|d_f - d_t|$, is 6.03 percentage points (pp). The benchmark's bound is 5 pp, and the checkpoint is rejected. We keep every trained weight byte-identical and recompute only the running statistics of its batch-normalization layers on retained training images. The same criterion then reads 0.26 pp, and the benchmark would admit the checkpoint on all ten of our refits.

The verdict moved because a batch-normalized checkpoint ships data-fitted statistics alongside its weights. Each batch-normalization layer stores a running mean and variance (Ioffe & Szegedy, 2015), which are fitted from data and applied at inference, although no gradient step writes them. A model retrained without the removed data, the reference that unlearning is judged against, fits these statistics on retained data by construction, because retained data is all it ever saw. An unlearned checkpoint ships whatever its own procedure left, such as a running average accumulated while its weights moved. Our refit gives the unlearned checkpoint statistics fitted on retained data, as a retrained model's are. Neither release we audit states which data, image transform or averaging rule produced the shipped values. Of 19 papers in a rule-based sample of the literature that audit such models against a retrained reference, only 4 mention these statistics (Section 4.4).

A number that moves under a recomputation of the statistics has two candidate explanations. In a deletion failure, removed data survives in the shipped statistics. With stale statistics, the shipped

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.

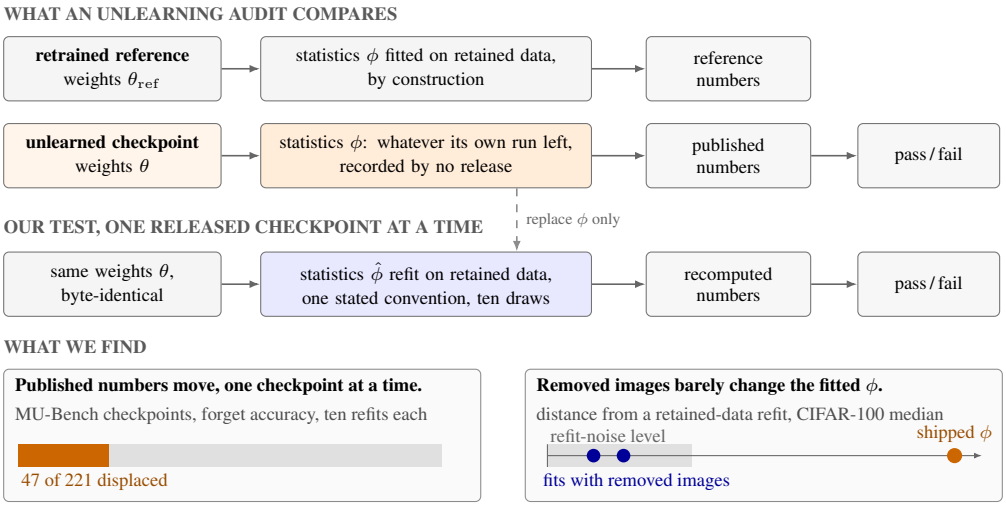

Figure 1: **Published unlearning numbers depend on how the batch-normalization statistics ϕ were fitted.** An audit's two models carry ϕ fitted differently. Our test refits only ϕ on retained data, at byte-identical weights θ. Bottom right (CIFAR-100, median of the 20 tested checkpoints): fits with removed images stay inside the grey band, where our refit reproduces the statistics of Unlearning Comparator's retrained references, while shipped ϕ sits far beyond it.

values no longer match the weights they ship with. Recomputing batch-normalization statistics at fixed weights is routine (Izmailov et al., 2018; Wu & Johnson, 2021), and concurrent work by Kalani et al. (2026) shows, on models its authors train themselves, that the operation can reverse an unlearning verdict. Two questions about tables the field already holds remain open: whether the published numbers of third-party checkpoints survive the operation, and which explanation applies when one moves.

We answer both with a refit test, which rebuilds each published number of a released checkpoint under one stated convention (Figure 1). It first reproduces the numbers from the release, then holds the trained weights byte-identical, refits only the normalization statistics on retained data, and recomputes the numbers and the release's own verdict. A refit depends on which retained images it draws, and we repeat it ten times with independent draws and decide each checkpoint on its own ten results. To test the deletion-failure explanation, we also swap removed images into the refit and measure what changes.

To count such movements, we call a checkpoint *displaced* when all ten of its refits raise an accuracy by more than a margin of 1.2 pp, the median seed-to-seed standard deviation that MU-Bench publishes. Forget accuracy is displaced on 47 of the 221 reproducible MU-Bench checkpoints we decide this way (Table 1).

A method's average shift does not identify which of its checkpoints are displaced. The average shift of NegGrad (gradient ascent on the removed data) on CIFAR-100 lies inside the margin, yet 8 of its 45 admitted checkpoints there are displaced (Figure 2). In a targeted test, removed data surviving in the statistics does not explain which checkpoints move either. On each MU-Bench dataset and architecture, we add removed images to the refit of the ten checkpoints the refit raises most and the ten it raises least. If removed data in the fitted statistics drove the shift, this should change the published numbers of the checkpoints that move. It barely changes the fitted statistics at the median. At their training-set share, removed images move forget accuracy by more than 0.5 pponly on four random-label checkpoints of CIFAR-10 ResNet-50, three of them among those the refit raises least. On CIFAR-100, the shipped statistics of the ten raised most instead sit farther from a fresh refit than those of the ten raised least, consistent with statistics accumulated while the weights were still changing (Section 4.2).

The convention also changes a few pass/fail verdicts. Of the 16 verdicts the refit flips on one draw, we call 4 *decisive*: their criterion value ends past the bound by more than twice what the refit does to models that no unlearning run touched. All four are on CIFAR-10, and two of them can be

checked against a printed criterion value. A published number on a batch-normalized checkpoint therefore describes that checkpoint under the statistics it shipped, and it should be quoted with its fitting convention.

Contributions.

- A refit test: reproduce published numbers, refit only normalization statistics at byte-identical weights, and decide each checkpoint by an exact sign test (Section 3).
- Forget accuracy rises past the seed spread on 47 of 221 MU-Bench checkpoints, even where a method's average shift lies inside it (Section 4.1).
- In a targeted test on 60 MU-Bench checkpoints, removed data in the statistics does not explain which of them move (Section 4.2).
- The convention changes pass/fail verdicts: of 16 that the refit flips, 4 are decisive, all on CIFAR-10 (Section 4.3).

2 Related work

Unlearning audits certify from outputs and weights. Evaluations compare an unlearned model with a retrained reference on accuracy over removed and retained data, often beside a membership-inference score (Golatkar et al., 2020; Shokri et al., 2017; Triantafillou et al., 2024). Later work changes the test statistic, the attack, the choice of removed data or the enforcement setting (Hayes et al., 2025; Wichert & Sikdar, 2024; Kim et al., 2026; Li et al., 2026). Thudi et al. (2022b) argue that a weight vector cannot certify unlearning, because other training histories reach the same weights, and Yu et al. (2025) show retrain equivalence unattainable once training has several stages. Both lines study what the weights can and cannot certify. The normalization statistics sit outside the weights, and every audit metric is computed through them. This paper keeps the metrics fixed and varies only those statistics (Appendix N places it against each line).

Recomputing normalization statistics at fixed weights. Recomputing batch-normalization statistics at fixed weights is standard after weight averaging (Izmailov et al., 2018) and after quantization (Nagel et al., 2019). Wu & Johnson (2021) show that the estimator used for these statistics changes accuracy, and their exact recomputation is the refit we use. Test-time adaptation re-estimates the same statistics on shifted data (Li et al., 2018; Schneider et al., 2020; Nado et al., 2020; Wang et al., 2021). In these uses the statistics are re-estimated to improve a model, and the variable of interest is the estimator or the data distribution, which our audit holds fixed.

Concurrent work. An unlearned model can be pushed to recover removed behavior by quantization (Zhang et al., 2025; Khan & Sohel, 2026), by fine-tuning on retained data (Siddiqui et al., 2025) and by recalibrating its normalization statistics (Jawandhia et al., 2026; Kalani et al., 2026). Kalani et al. (2026) formalize the same weight-preserving recalibration and show, on nine methods they train themselves, that it can reverse forget accuracy by up to 78 pp. Jawandhia et al. (2026) study six self-trained models. Our question is whether the tables already published for third-party checkpoints survive the refit, checkpoint by checkpoint, and whether removed data in the statistics explains which checkpoints move (Appendices N and O).

3 A refit test for released checkpoints

The refit. Our setting is batch-normalized convolutional classifiers, the models that ship running statistics. Let θ be a checkpoint's trained weights and ϕ its normalization statistics, the per-channel running mean and variance of each batch-normalization layer. A published number is $m(\theta, \phi)$, one of d_f, d_r and d_t. The refit replaces ϕ with $\hat{\phi}$, computed at fixed θ from 8000 images of the release's own retained training data. We report the shift $m(\theta, \hat{\phi}) - m(\theta, \phi)$ in pp and the accuracies themselves in percent. The statistics are exact averages computed layer by layer (Algorithm 1), which makes $\hat{\phi}$ depend only on θ, on which images are drawn and on their augmentation (Appendix C).

Table 1: **Every population, how many of its checkpoints reproduce their published numbers, and how many the refit displaces.** *Checkpoints* counts those measured (for MU-Bench CIFAR-100, the 72 its criterion admits); *reproduced* counts those reproducing every published accuracy within 2 pp; *decided* counts those entering the ten-refit decision. *Displaced* means a gain beyond the margin on all ten refits, with the false-discovery rate controlled across the 221 decided checkpoints (Section 3). Released checkpoints use the 1.2 pp margin, and each self-trained population uses its own. SVHN is decided on its four checkpoints that pass the criterion and a utility floor, and self-trained CIFAR-100 on the 45 trained as the implementation ships them (Section 4.4). Unlearning Comparator is analyzed in pairs instead (Section 3).

						displaced		
release	data	model	checkpoints	reproduced	decided	d_f	d_r	d_t
MU-Bench	CIFAR-100	ResNet-50	72	71	71	20	16	15
MU-Bench	CIFAR-10	ResNet-50	165	120	120	16	14	15
MU-Bench	CIFAR-10	ResNet-34	40	30	30	11	9	10
Unlearning Comparator	CIFAR-10	ResNet-18	50	42	–	–	–	–
released, total				263	221	47	39	40
self-trained	SVHN	VGG-16-BN	40	–	4	2	–	–
self-trained	CIFAR-100	VGG-16-BN	99	–	45	20	20	22

The images pass through the training transform of the code that produced the checkpoint, the transform under which the shipped statistics were accumulated. Refitting MU-Bench's CIFAR-100 original model, the model before unlearning, under the evaluation transform instead shifts its test accuracy by -2.55 pp, against -0.08 pp under the training transform (Appendix C). The transform is therefore part of the convention a release should state.

The refit recovers retained-data statistics where they are known. Unlearning Comparator ships ten retrained references whose statistics were accumulated on retained data during training. We measure the distance between two sets of statistics as the median, over all channels, of the absolute difference in running means divided by the refit's per-channel standard deviation. Our refit of those references lands at a median distance of 0.0019 from their shipped statistics, and we call this distance the *refit-noise level* and use it for all groups. The release's unlearned checkpoints sit $1.8\times$ to $47.1\times$ that distance from their own refits, depending on the method (Appendix C).

Populations. We audit the two public releases we found that publish a per-checkpoint table an outsider can reproduce (Table 1). MU-Bench (Cheng & Amiri, 2024) releases CIFAR-100 ResNet-50 checkpoints for four methods: bad teaching (Chundawat et al., 2023), NegGrad, random label (Graves et al., 2021) and SalUn (Fan et al., 2024). They vary the procedure (a base version and its curriculum and second-order variants), the removal ratio (the share of the training set removed, from 2% to 10%) and the seed, and only some combinations are released. MU-Bench also releases CIFAR-10 checkpoints on ResNet-50 and ResNet-34 (He et al., 2016), giving three groups of one dataset and architecture each. Its criterion admits a checkpoint when $|d_f - d_t| \leq 5$ pp. On CIFAR-100 we measure the checkpoints it admits and use the ones it excludes only for verdict changes, and on CIFAR-10 we measure all of them.

Unlearning Comparator (Lee et al., 2026) releases CIFAR-10 ResNet-18 checkpoints that each delete one class, for four methods (CF-k (Goel et al., 2022), AdvNegGrad, SalUn and SSD (Foster et al., 2024)), and a retrained reference for each of the ten classes. Its criterion requires $d_r \geq 95\%$ and $d_f \leq 10\%$ on the training split, values its published table omits. A checkpoint enters our counts only if it reproduces each accuracy its release publishes for it within 2 pp with its shipped statistics, and shifts are measured from our reproduction.

SalUn checkpoints mostly fail that rule. Their forget accuracy fails to reproduce on all 55 MU-Bench CIFAR-10 SalUn checkpoints, 8 of the 10 on Unlearning Comparator fail, and MU-Bench's criterion admits none of SalUn's CIFAR-100 checkpoints. Our MU-Bench counts therefore cover three of its four methods. Two self-trained populations, pooled with neither release, test how far the result reaches (Section 4.4).

Deciding one checkpoint. A single refit draws one retained sample and one augmentation stream, and its shift carries that sampling noise. We therefore refit each MU-Bench checkpoint ten times with both drawn independently, and each Unlearning Comparator checkpoint three times, paired with the retrained reference for the same deleted class. Three rules turn these shifts into decisions (Appendix D).

- **Displaced checkpoints.** A checkpoint is displaced on an accuracy when all ten shifts exceed $+1.2$ pp. The test is one-sided, for gains. If the checkpoint's median shift sat exactly at the margin, ten shifts above it would occur with probability 0.00098, an exact sign test that is distribution-free. We control the false-discovery rate across the decided checkpoints with the Benjamini–Yekutieli procedure at $q = 0.05$, which is valid under any dependence between them. The ten refits vary only the refit's own sampling, and seed-to-seed variation enters only through the margin. The margin compares a shift within one checkpoint with the spread between training seeds, which a reader already accepts as noise in the release's table.
- **Method averages.** For a method's average shift we use two one-sided tests at the 90% level against the same margin, with cluster-robust intervals over MU-Bench's three procedure variants (Schuirmann, 1987; Lakens, 2017). An interval inside ± 1.2 pp makes the average equivalent to zero at that margin, an interval outside it makes the average different from zero, and an interval spanning the margin leaves it unresolved.
- **Flipped and decisive verdicts.** A verdict flips when the recomputed numbers change a release's own pass/fail verdict. A baseline b sets the scale of a flip. It is how far the refit moves the criterion value of models no unlearning run touched, MU-Bench's original model and Unlearning Comparator's retrained references. For each accuracy j the criterion uses, let $\bar{\Delta}_j$ and s_j be the mean and standard deviation of its shifts across refits (across the ten references on Unlearning Comparator). Then

  $$b = \textstyle\sum_j \big(|\bar{\Delta}_j| + 2\max(s_j,\, 0.10\,\text{pp})\big).$$

  A flipped verdict is decisive when it lands more than $2b$ past the bound. On MU-Bench CIFAR-100, b is 0.70 pp (Table 3 lists each group's value).

4 Results

4.1 Published numbers move, checkpoint by checkpoint

Refitting the statistics moves published numbers on individual checkpoints by more than the release's seed-to-seed spread, and a method's average does not predict which checkpoints move. Of the MU-Bench checkpoints we decide with ten refits, 47 are displaced on forget accuracy, 39 on retain accuracy and 40 on test accuracy (Table 1), and all three counts survive the false-discovery control.

The displaced shifts are large next to the refit's own sampling noise. On CIFAR-100, the standard deviation across one checkpoint's refits has a median of 0.18 pp, and the original model, from which nothing was removed, gains only $+0.21$ pp of forget accuracy.

NegGrad on MU-Bench CIFAR-100 has an average shift inside the margin and still has displaced checkpoints (Figure 2). Its average forget-accuracy shift has a 90% interval that excludes zero but lies inside the margin, which makes the average negligible at that margin, and we find 8 of its 45 admitted checkpoints individually displaced.

Bad teaching's average shift cannot be resolved, because it depends strongly on the procedure variant. Its forget-accuracy shifts average $+2.47$ pp, yet with MU-Bench's three procedure variants as clusters, its interval would span the margin even if the true mean were zero (Appendix E). By contrast, a single bad-teaching checkpoint shifts $+3.83$ pp with a spread of only 0.06 pp across its refits.

Unlearning Comparator ships the retrained reference itself, which lets the refit act on both sides of an audit. We pair each method with the reference for the same deleted class and measure retain test accuracy, on held-out images of the retained classes. The reference and CF-k shift it by amounts

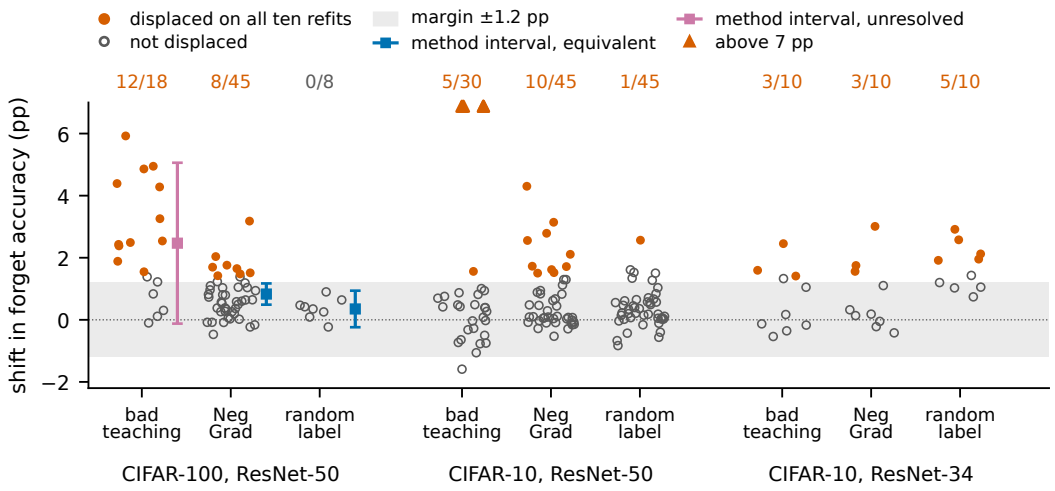

Figure 2: **Shifts vary within each method, and a method-level interval does not predict them.** Each dot is one released MU-Bench checkpoint's forget-accuracy shift, averaged over its ten refits; filled dots are displaced on all ten, and the fraction above each column counts them. The grey band is the ± 1.2 pp margin. Squares with bars are the method-level 90% intervals on CIFAR-100: NegGrad's lies inside the margin, so its average shift is negligible at that margin, yet 8 of its 45 checkpoints are displaced. Bad teaching's interval spans the margin and cannot be resolved.

equivalent to zero, and AdvNegGrad by $+1.92$ pp, with an interval outside the margin (Table 5). SSD's average shift of $+11.38$ pp comes from three checkpoints that fail their release's own criterion (Appendix B).

The counts fall monotonically as the margin grows, on all three accuracies, and halving or doubling the retained pool moves no CIFAR-100 checkpoint's shift by more than the margin (Appendix D). The movement also appears on a second metric, the zero-retrain-forgetting score of Chundawat et al. (2023). There, 21 of the 72 admitted checkpoints move past the release's own spread in one direction or the other, by an interval rule over three refits (Appendix M).

4.2 Removed data and which checkpoints move

Removed data surviving in the shipped statistics does not explain which checkpoints move. If it did, the movement would be a deletion failure, the explanation a reader reaches for first. We test it on 20 checkpoints in each of MU-Bench's three groups, the ten the refit raises most and the ten it raises least, by mean forget-accuracy shift. On CIFAR-100, 19 of the 20 come from bad teaching and NegGrad, whose released code updates the statistics on batches made only of removed images (Appendix K), a direct path for removed data into what they ship.

The test changes only which images the refit sees, and keeps the weights, estimator, transform and pool size fixed. Let k be the number of removed images that a uniform sample of the pool's size would contain. Two fits share the pool's other images, all retained. The retained-only fit, an ordinary refit, fills the k places with further retained images, and the fit with removed images puts removed ones there, at their training-set share. If removal does not matter, the two fits are exchangeable and a sign test on their difference is exact. A third fit takes all removed images, up to 5000, and fills the rest of the pool with retained images. It maximizes the removed share and serves as a stress test.

On two of the three groups, removed images barely move a published number. When we add them at their training-set share, they move forget accuracy by $+0.02$ pp on CIFAR-100 and by $+0.01$ pp on CIFAR-10 ResNet-34, on average (Table 2b). Each checkpoint's average change stays below 0.5 pp on both groups, even with all removed images in the fit, and the ResNet-34 set includes five random-label checkpoints, all of the base procedure variant.

On CIFAR-10 ResNet-50, removed data does move the published forget accuracy of four checkpoints, all random-label, a subset identified post hoc. At the training-set share they gain $+0.96$ to

Table 2: **At the median, the shipped statistics sit far from the retained-only fit, while removed images barely change the fitted statistics in any group. Removed images move forget accuracy only on CIFAR-10 ResNet-50.** Twenty checkpoints per MU-Bench group, the ten the refit raises most and the ten it raises least. Panel (a) gives distances between two fitted statistics (Section 3) and, in parentheses, how many checkpoints lie beyond the refit-noise level of 0.0019. Panel (b) gives the change in published forget accuracy against the retained-only fit, averaged over independent draws, with a threshold of 0.5 pp fixed in advance. Appendix F lists every tested CIFAR-10 ResNet-50 checkpoint.

MU-Bench group (20 checkpoints each)	CIFAR-100 ResNet-50	CIFAR-10 ResNet-50	CIFAR-10 ResNet-34
(a) median distance from the retained-only fit			
shipped statistics	0.0054 (20)	0.0044 (16)	0.0098 (20)
removed images, training-set share	0.0006 (0)	0.0006 (2)	0.0008 (0)
every removed image	0.0010 (1)	0.0009 (4)	0.0012 (1)
(b) change in forget accuracy d_f (pp)			
training-set share, mean	+0.02	+0.25	+0.01
largest absolute change	0.18	1.46	0.08
checkpoints over 0.5 pp	0	4	0
every removed image, mean	+0.06	+1.50	+0.04
largest absolute change	0.27	9.19	0.16
draws per checkpoint	6	13	13

+1.46 pp, and two of them reach the margin. With all removed images in the fit, the largest gain is +9.19 pp. Three of the four sit among the ten the refit raises least.

The checkpoints the refit raises most respond far less. These four bad-teaching models, which the refit raises by +11.28 to +19.94 pp, change by at most 0.07 pp when removed images enter the fit at their training-set share.

At the median, removed images barely register in the fitted statistics themselves. In each group, both fits with removed images sit within the refit-noise level of the retained-only fit at the median (Table 2a). On CIFAR-10 ResNet-50, 2 of 20 lie beyond it at the training-set share, and the median therefore misses the forget-accuracy effect there. We treat both distances as upper bounds on what removed images do to the statistics, because swapping k images for k others perturbs a finite-sample estimate even when their source is irrelevant.

The shipped statistics sit farther away, and on CIFAR-100 farther for the checkpoints the refit raises most. There they lie beyond the refit-noise level on all tested checkpoints, at a median distance of 0.0081 for the ten raised most against 0.0029 for the ten raised least. Adding removed images moves the fit by 0.0006 in both (Table 10). This association is consistent with statistics accumulated while the weights moved and never refit at the final weights (Appendix F).

4.3 Pass/fail verdicts and method rankings

Decided on one refit, the refit flips 16 pass/fail verdicts, and 4 of these flips are decisive (Table 3). The refit is a second defensible convention for the same weights. A flipped verdict therefore shows that the published verdict depends on a choice the release does not record. It does not show that the benchmark erred, since scoring shipped statistics is also defensible. We apply two checks to each flip: whether it survives when judged against the release's own printed number in place of our reproduction, and whether it is decisive.

On MU-Bench CIFAR-100, verdicts flip only near the bound. All 12 flipped verdicts there land within the refit's baseline of the bound, or just beyond it, and far short of twice the baseline (Figure 5a in Appendix G). These flips are comparable to the movement the refit produces with nothing removed, which indicates threshold effects.

The four decisive flips lie on CIFAR-10. On MU-Bench CIFAR-10 ResNet-50, the checkpoint from the introduction (the second-order variant at seed 87 and a 10% removal ratio) and a second checkpoint that the criterion rejects land deep inside the bound after the refit. Both flip on the

Table 3: **Of 16 verdicts the refit flips, 4 are decisive, two on each release and none on MU-Bench CIFAR-100.** *Flipped*: verdicts that change when decided on our recomputed numbers. *Out / in*: checkpoints leaving or entering the admitted set (for Unlearning Comparator, failing checkpoints that pass). *Holds on published*: the flip survives when the value before the refit is the release's own printed number instead of our reproduction of it; Unlearning Comparator's published table does not include the quantities its criterion uses. *Baseline*: how far the refit moves the criterion of models no unlearning run touched (pp). *Decisive*: past the bound by more than twice the baseline. Appendix G lists all 16, with ten-refit stability for the 10 among the 221 decided checkpoints.

release group	out / in (count)	flipped	holds on published	baseline (pp)	decisive
MU-Bench, CIFAR-100, ResNet-50	8 / 4	12	10	0.70	0
MU-Bench, CIFAR-10, ResNet-50	0 / 2	2	2	0.61	2
MU-Bench, CIFAR-10, ResNet-34	0 / 0	0	0	0.69	0
Unlearning Comparator	0 / 2	2	–	0.20	2
total		16	12		4

published number too and on all ten refits, the only two flips that pass both checks. On Unlearning Comparator, a SalUn and an SSD checkpoint that fail the retain-accuracy bar pass after the refit, while their forget accuracy stays at or below 0.18%.

MU-Bench CIFAR-100 method rankings by forget accuracy survive the refit. Of 32 comparisons we make between two such methods at the same procedure variant, removal ratio and seed, none reverses order on forget accuracy, while 3 reverse on test accuracy.

4.4 How far the result reaches

Displacement recurs on two self-trained populations, including one built so that a verdict could flip.

A population built near the criterion. The first self-trained population places its checkpoints near the criterion's bound, where few released checkpoints sit, to make a flipped verdict reachable. We trained CIFAR-100 VGG-16-BN models (Simonyan & Zisserman, 2015) with a public implementation (Fan et al., 2024) at its own defaults, changing only the optimization strength, which we set so that the gap $|d_f - d_t|$ lands near 5 pp. We count only its 45 checkpoints trained as the implementation ships them.

Shifts there are as large as in the releases, with 20 checkpoints rising past that population's own margin and one by $+13.26$ pp. No verdict flips, and the nearest checkpoint comes within 0.20 pp of flipping. Most checkpoints sit above the bound, where only a narrowing gap could flip a verdict, and the refit widens that gap on most of them (Appendix H). The verdict claim therefore rests on the two releases.

On the second self-trained population, SVHN with VGG-16-BN (Netzer et al., 2011), the displacement reappears at a smaller scale. SalUn is the one method there whose updates mix removed and retained data. It shifts by $+0.47$ pp on its four checkpoints that pass the criterion and a utility floor (Appendix H), and two of them rise past that population's own margin (Appendix H).

Other fitted states and forgetting quality. Static quantization scales are a fitted state that any architecture can carry. Calibrated on retained data, they displace none of 118 MU-Bench CIFAR-100 checkpoints, ResNet-50 and Swin models from the same release. The design reliably detects only shifts of 2.0 pp or more. Only calibration entirely on removed images, a stress test, moves three ResNet-50 models past the margin (Appendix I). On MU-Bench CIFAR-100, forgetting quality has no usable readout. The standard membership attack is at chance, and the original model drifts further under the refit than the unlearned checkpoints, which voids a readout based on drift toward the original model. On Unlearning Comparator, the refit moves one method, SSD, $+0.0167$ of the way back toward the original model in its confidence on the deleted class, in the same direction on all ten classes (Appendix J).

Recording and screening the convention. The literature seldom records the convention. Of 42 unlearning papers selected by a rule fixed in advance (Appendix L), 19 run batch-normalized convolutional models against a retrained reference, and only 4 of those mention the normalization statistics at all.

We also test quantities readable from released files as screens, that is, as predictors of which checkpoints are displaced. Two of them, the distance between a checkpoint's statistics and the original model's and the batch count each layer records, do not predict the shift within any method. The relative L_2 distance of the trained weights from the original model does better. Its rank correlation with the mean forget-accuracy shift is significantly positive in 4 of the 9 combinations of MU-Bench method and release group, all of them on CIFAR-10 (Appendix K).

At the method level, the released code separates methods by how far they shift. Some released code updates the statistics on batches made only of removed data. We find that methods with such code shift 1.86 pp more on average than CF-k and the retrained reference, when we pool forget accuracy on MU-Bench with retain test accuracy on Unlearning Comparator. A test of this ordering gives $p = 0.056$, the smallest value possible with these method counts. That ordering identifies neither which checkpoint within a method moves nor whether removed data in the statistics moves it (Appendix K).

5 Limitations

We establish the result on batch-normalized convolutional classifiers, in two releases on CIFAR-10 and CIFAR-100 and two self-trained populations. Layer- and RMS-normalized models ship no running statistics. Among other fitted states, quantization scales showed no displacement at one operating point, where shifts between the 1.2 pp margin and 2.0 pp are not ruled out, and temperatures, feature scalers and class priors are untested. Counts are rates within the checkpoints that reproduce their published numbers, and on MU-Bench CIFAR-100 within those its criterion admits. SalUn, whose Unlearning Comparator code updates its statistics on batches made only of removed data, is absent from the MU-Bench counts and from the removed-image test. That test covers 20 checkpoints per MU-Bench group at their final weights. On CIFAR-10 ResNet-50 it finds removed images moving the forget accuracy of the four random-label checkpoints tested, a response whose cause is not identified. On CIFAR-100, the distance of shipped statistics from a refit is a descriptive association with which checkpoints move, with no procedural cause identified. The ten-refit decision controls the false-discovery rate, not the familywise error, and establishes the direction of a shift only. Shifts are measured relative to one refit convention, which is why we recommend naming a convention and do not prescribe one. The verdict result is an existence result on four checkpoints, two of them checkable against a printed criterion value and all on CIFAR-10, and the forgetting-quality effect is one method on ten classes sharing one original model.

6 Conclusion

A published unlearning number on a batch-normalized checkpoint depends on how that checkpoint's normalization statistics were fitted, and no release we audit records how. Refitting an unlearned checkpoint's statistics on retained data, as a retrained reference's are fitted, at byte-identical weights moves published numbers checkpoint by checkpoint, and, in a targeted test, removed data in the statistics does not explain which checkpoints move. The refit supports claims about how reproducible audit numbers are, not about deletion safety. Two practices follow. An auditor holding one checkpoint needs that checkpoint's own number, because a method-level result bounds only an average, and a refit measures it in under an hour on one GPU. A release can remove the ambiguity by stating the convention beside each audit number, for example: *statistics as shipped; a refit on 8000 retained images under the training transform moves* $|d_f - d_t|$ *from* 6.03 *to* 0.26 *pp*.

AI USE STATEMENT

AI assistants based on large language models were used for part of the execution of this work, under the authors' direction: implementing the measurement code and the loaders for the third-party checkpoint releases. All research decisions were made by the authors, who verified every AI-assisted output and every reported result. The authors take full responsibility for the content of this paper.

ETHICS STATEMENT

This work audits publicly released model checkpoints and their published evaluation tables. It involves no human subjects, no new data collection and no personal data. It reports that one released method's published forget accuracy cannot be reproduced under the conventions we could recover. It also reports that other published numbers, while reproducible as shipped, change under a benign refit of the statistics those checkpoints ship. Both are statements about measurement validity, not about the intent or competence of the releases' authors. On Unlearning Comparator checkpoints that fail their own criterion, the refit restores retain accuracy while forget accuracy stays near zero. Deletion audits are used to substantiate legal erasure obligations (Fosch Villaronga et al., 2018; Ginart et al., 2019), so a party relying on an audit verdict benefits from knowing the convention it was computed under.

REPRODUCIBILITY STATEMENT

The audited objects are public third-party releases, so every result can be checked independently of our code. Section 3 states the refit, the reproduction rule and the decision rule; Appendix C gives the refit as an algorithm with its transforms, pool size and seeds, and Appendix B the populations and reproduction rates. The anonymized code supplement contains the refit implementation and loaders for both releases with the identifiers of every checkpoint used. It also holds the analysis scripts that produce each reported aggregate, the aggregate result files, and one script per table and figure that regenerates it from those files. With it, a reader can take one released checkpoint, confirm that it reproduces its published numbers, refit its statistics, and recompute its numbers and verdict. Refitting 15 checkpoints once took 3368 seconds on a slice of one A100 GPU, so ten refits of one checkpoint cost under 40 GPU-minutes.

REFERENCES

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE Symposium on Security and Privacy (SP)*, 2021.

Xavier Bouthillier, Pierre Delaunay, Mirko Bronzi, Assya Trofimov, Brennan Nichyporuk, Justin Szeto, Nazanin Mohammadi Sepahvand, Edward Raff, Kanika Madan, Vikram Voleti, Samira Ebrahimi Kahou, Vincent Michalski, Tal Arbel, Chris Pal, Gaël Varoquaux, and Pascal Vincent. Accounting for variance in machine learning benchmarks. In *Conference on Machine Learning and Systems (MLSys)*, 2021.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *IEEE Symposium on Security and Privacy (SP)*, 2015.

Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramèr. Membership inference attacks from first principles. In *IEEE Symposium on Security and Privacy (SP)*, 2022.

Jiali Cheng and Hadi Amiri. MU-Bench: A multitask multimodal benchmark for machine unlearning. *arXiv preprint arXiv:2406.14796*, 2024.

Xinwen Cheng, Zhehao Huang, Wenxing Zhou, Zhengbao He, Ruikai Yang, Yingwen Wu, and Xiaolin Huang. Remaining-data-free machine unlearning by suppressing sample contribution. In *International Conference on Learning Representations (ICLR)*, 2026.

Somnath Basu Roy Chowdhury, Rahul Kidambi, Avinava Dubey, David Wang, Gokhan Mergen, Amr Ahmed, and Aranyak Mehta. Inference-time unlearning using conformal prediction. *arXiv preprint arXiv:2602.03787*, 2026.

Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan Kankanhalli. Can bad teaching induce forgetting? unlearning in deep networks using an incompetent teacher. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2023.

Andrea D'Angelo, Claudio Savelli, Gabriele Tagliente, Flavio Giobergia, Elena Baralis, and Giovanni Stilo. How to make reproducible research in machine unlearning with ERASURE. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2025.

Zahra Dehghani, Pablo Piantanida, and Mohammadhadi Shateri. Source-free class relearning: Diagnosing forgetting in class unlearning. *arXiv preprint arXiv:2609.02018*, 2026.

Ngoc Phu Doan, Chongyan Gu, and Ihsen Alouani. Batch normalization amplifies memorization and privacy risks. *arXiv preprint arXiv:2605.24420*, 2026.

Chongyu Fan, Jiancheng Liu, Yihua Zhang, Eric Wong, Dennis Wei, and Sijia Liu. SalUn: Empowering machine unlearning via gradient-based weight saliency in both image classification and generation. In *International Conference on Learning Representations (ICLR)*, 2024.

Eduard Fosch Villaronga, Peter Kieseberg, and Tiffany Li. Humans forget, machines remember: Artificial intelligence and the right to be forgotten. *Computer Law & Security Review*, 2018.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2024.

Jonathan Frankle, David J. Schwab, and Ari S. Morcos. Training BatchNorm and only BatchNorm: On the expressive power of random features in CNNs. In *International Conference on Learning Representations (ICLR)*, 2021.

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making AI forget you: Data deletion in machine learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Shashwat Goel, Ameya Prabhu, Amartya Sanyal, Ser-Nam Lim, Philip Torr, and Ponnurangam Kumaraguru. Towards adversarial evaluations for inexact machine unlearning. *arXiv preprint arXiv:2201.06640*, 2022.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens van der Maaten. Certified data removal from machine learning models. In *International Conference on Machine Learning (ICML)*, 2020.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy. In *IEEE Conference on Secure and Trustworthy Machine Learning (SaTML)*, 2025.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015.

Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2018.

Vedant Jawandhia, Daksh Ahuja, Ghufran Alam Siddiqui, Prashant Trivedi, Yash Sinha, and Pratik Narang. PURGE: Projected unlearning via retain-guided erasure. *arXiv preprint arXiv:2606.03808*, 2026.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Aaryaman Kalani, Murari Mandal, Dhruv Kumar, Mohan Kankanhalli, and Yash Sinha. The BatchNorm illusion: Diagnosing normalization artifacts in machine unlearning evaluation. *arXiv preprint arXiv:2609.08901*, 2026.

Abdullah Ahmad Khan and Ferdous Sohel. DurableUn: Quantization-induced recovery attacks in machine unlearning. *arXiv preprint arXiv:2605.02196*, 2026.

Yongwoo Kim, Sungmin Cha, and Donghyun Kim. Are we truly forgetting? a critical re-examination of machine unlearning evaluation protocols. *Engineering Applications of Artificial Intelligence*, 2026.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023.

Daniël Lakens. Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. *Social Psychological and Personality Science*, 2017.

Jaeung Lee, Suhyeon Yu, Yurim Jang, Simon S. Woo, and Jaemin Jo. Unlearning comparator: A visual analytics system for comparative evaluation of machine unlearning methods. *IEEE Transactions on Visualization and Computer Graphics*, 2026.

Jiazhuo Li, Yu Zhang, Yiming Fei, Kangkang Dong, Xiaojun Zhu, Houde Liu, and Jinze Tao. Rethinking demonstration unlearning in imitation learning for robotics. *arXiv preprint arXiv:2608.20784*, 2026.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*, 2024.

Yanghao Li, Naiyan Wang, Jianping Shi, Xiaodi Hou, and Jiaying Liu. Adaptive batch normalization for practical domain adaptation. *Pattern Recognition*, 2018.

Qinqi Lin, Ningning Ding, Lingjie Duan, and Jianwei Huang. Governing AI forgetting: Auditing for machine unlearning compliance. *arXiv preprint arXiv:2602.14553*, 2026.

Mario Lucic, Karol Kurach, Marcin Michalski, Sylvain Gelly, and Olivier Bousquet. Are GANs created equal? a large-scale study. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary C. Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *Conference on Language Modeling (COLM)*, 2024.

Zachary Nado, Shreyas Padhy, D. Sculley, Alexander D'Amour, Balaji Lakshminarayanan, and Jasper Snoek. Evaluating prediction-time batch normalization for robustness under covariate shift. *arXiv preprint arXiv:2006.10963*, 2020.

Markus Nagel, Mart van Baalen, Tijmen Blankevoort, and Max Welling. Data-free quantization through weight equalization and bias correction. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

Aviraj Newatia, Michael Cooper, Viet Nguyen, and Rahul G. Krishnan. Mitigating privacy risk via forget set-free unlearning. In *International Conference on Learning Representations (ICLR)*, 2026.

Joelle Pineau, Philippe Vincent-Lamarre, Koustuv Sinha, Vincent Larivière, Alina Beygelzimer, Florence d'Alché Buc, Emily Fox, and Hugo Larochelle. Improving reproducibility in machine learning research: A report from the NeurIPS 2019 reproducibility program. *Journal of Machine Learning Research*, 2021.

Edward Raff. A step toward quantifying independently reproducible machine learning research. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018.

Sahasrajit Sarmasarkar, Anastasia Koloskova, and Sanmi Koyejo. Auditing of unlearning algorithms. *arXiv preprint arXiv:2607.05898*, 2026.

Steffen Schneider, Evgenia Rusak, Luisa Eck, Oliver Bringmann, Wieland Brendel, and Matthias Bethge. Improving robustness against common corruptions by covariate shift adaptation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Donald J. Schuirmann. A comparison of the two one-sided tests procedure and the power approach for assessing the equivalence of average bioavailability. *Journal of Pharmacokinetics and Biopharmaceutics*, 1987.

Seonguk Seo, Dongwan Kim, and Bohyung Han. Revisiting machine unlearning with dimensional alignment. In *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2025.

Weijia Shi, Jaechan Lee, Yangsibo Huang, Sadhika Malladi, Jieyu Zhao, Ari Holtzman, Daogao Liu, Luke Zettlemoyer, Noah A. Smith, and Chiyuan Zhang. MUSE: Machine unlearning six-way evaluation for language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Yingdan Shi, Sijia Liu, Kaize Ding, and Ren Wang. Tackling fake forgetting through uncertainty quantification. In *International Conference on Machine Learning (ICML)*, 2026.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (SP)*, 2017.

Shoaib Ahmed Siddiqui, Adrian Weller, David Krueger, Gintare Karolina Dziugaite, Michael C. Mozer, and Eleni Triantafillou. From dormant to deleted: Tamper-resistant unlearning through weight-space regularization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.

David M. Sommer, Liwei Song, Sameer Wagh, and Prateek Mittal. Athena: Probabilistic verification of machine unlearning. *Proceedings on Privacy Enhancing Technologies*, 2022.

Zirui Song, Huaxing Liu, Xiang Wang, Shuai Li, Xinye Li, Lang Gao, Jinghui Zhang, Zheng Lu, Fengxian Ji, Xiaojun Chang, and Xiuying Chen. Measure, don't optimize: Forecasting recovery in LLM unlearning. *arXiv preprint arXiv:2608.11408*, 2026.

Anvith Thudi, Gabriel Deza, Varun Chandrasekaran, and Nicolas Papernot. Unrolling SGD: Understanding factors influencing machine unlearning. In *IEEE European Symposium on Security and Privacy (EuroS&P)*, 2022a.

Anvith Thudi, Hengrui Jia, Ilia Shumailov, and Nicolas Papernot. On the necessity of auditable algorithmic definitions for machine unlearning. In *USENIX Security Symposium (USENIX Security)*, 2022b.

Eleni Triantafillou, Peter Kairouz, Fabian Pedregosa, Jamie Hayes, Meghdad Kurmanji, Kairan Zhao, Vincent Dumoulin, Julio Jacques Junior, Ioannis Mitliagkas, Jun Wan, Lisheng Sun Hosoya, Sergio Escalera, Gintare Karolina Dziugaite, Peter Triantafillou, and Isabelle Guyon. Are we making progress in unlearning? findings from the first NeurIPS unlearning competition. *arXiv preprint arXiv:2406.09073*, 2024.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations (ICLR)*, 2021.

Leon Wichert and Sandipan Sikdar. Rethinking evaluation methods for machine unlearning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, 2024.

Yuxin Wu and Justin Johnson. Rethinking "batch" in BatchNorm. *arXiv preprint arXiv:2105.07576*, 2021.

Cheng Xu, Nan Yan, Liming Chen, and M-Tahar Kechadi. Phantom gains: Auditing self-improvement against a measured null. *arXiv preprint arXiv:2608.20290*, 2026.

Jiatong Yu, Yinghui He, Anirudh Goyal, and Sanjeev Arora. On the impossibility of retrain equivalence in machine unlearning. *arXiv preprint arXiv:2510.16629*, 2025.

Zhiwei Zhang, Fali Wang, Xiaomin Li, Zongyu Wu, Xianfeng Tang, Hui Liu, Qi He, Wenpeng Yin, and Suhang Wang. Catastrophic failure of LLM unlearning via quantization. In *International Conference on Learning Representations (ICLR)*, 2025.

Kairan Zhao, Meghdad Kurmanji, George-Octavian Bărbulescu, Eleni Triantafillou, and Peter Triantafillou. What makes unlearning hard and what to do about it. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Chencheng Zhu. Reading the gate, not the interference: Output-side interference measurement does not track merge collapse. *arXiv preprint arXiv:2608.11797*, 2026.

A TERMS USED IN THE MAIN TEXT

Each term is defined where it first appears in the main text; this list collects the definitions.

Refit. Replacing a checkpoint's batch-normalization statistics with exact averages computed at its fixed weights from 8000 retained training images under the training transform (Section 3, Algorithm 1). The removed-image test also refits on image sets that include removed images (Section 4.2).

Shift. A published accuracy after the refit minus the same accuracy with the shipped statistics, in percentage points (pp).

Margin. 1.2 pp, the median standard deviation across training seeds that MU-Bench publishes. Each self-trained population has its own margin (Section 4.4).

Reproduced, decided. A checkpoint is reproduced when its shipped statistics give every accuracy its release publishes within 2 pp, and decided when it enters the ten-refit decision.

Displaced. A checkpoint whose ten independent refits all raise an accuracy by more than the margin; the counts control the false-discovery rate (Section 3). The *most* and *least displaced* are the checkpoints with the largest and smallest mean shift.

Equivalent, different, unresolved. A method's 90% interval on its average shift lies inside the margin, outside it, or across it (two one-sided tests).

Criterion. A release's own pass/fail rule: $|d_f - d_t| \leq 5$ pp on MU-Bench; $d_r \geq 95\%$ and $d_f \leq 10\%$ on Unlearning Comparator.

Flipped verdict. A pass/fail verdict that changes under the refit.

Baseline. How far the refit moves the criterion value of models no unlearning run touched (MU-Bench's original model; Unlearning Comparator's retrained references): summed over the accuracies the criterion uses, the absolute mean movement plus two standard deviations across refits, each standard deviation floored at 0.10 pp (Section 3).

Decisive. A flipped verdict that lands more than twice the baseline past the bound.

Original model. The model a release trained before any unlearning.

Retained-only fit and fits with removed images. Refits on 8000-image sets holding no removed images (the retained-only fit), removed images at their training-set share, or every removed image (Section 4.2).

Distance and refit-noise level. The median over channels of the absolute difference in running means between two sets of statistics, divided by the refit's running standard deviation. The refit-noise level, 0.0019, is the distance at which our refit reproduces the statistics of Unlearning Comparator's retrained references.

B Populations and reproduction

Every checkpoint enters an aggregate only after it reproduces its own published numbers with its shipped statistics: d_f, d_r and d_t each within 2 pp of the release's table (Table 4). MU-Bench releases 180 CIFAR-100 ResNet-50 checkpoints. Its own criterion admits 72; another 73 fail it while publishing usable values, and 35 publish no usable result file. We measure the 73 excluded ones only for verdicts that flip into the admitted set.

Two absolute checks tie our pipelines to the releases. MU-Bench's original model, before any unlearning, recomputes to a test accuracy of 83.28% against its published 83.28%, while two nearby preprocessing variants give 82.17% and 81.31%, so the check discriminates. On Unlearning Comparator the reproduction error is 0.02 pp, and refitting under permuted image order or a different batch size changes the recomputed accuracy by 0.00 pp.

SalUn fails to reproduce on both releases, in different ways. On MU-Bench CIFAR-10 all 55 of its checkpoints fail on forget accuracy, 54 of them on forget accuracy alone. Their retain and test accuracies reproduce to a median error of 0.04 pp and 0.01 pp, as closely as every other method, which fails 0 of 150. On Unlearning Comparator 8 of its 10 checkpoints fail, and the failures reach both published accuracies, 6 of 10 on forget and 6 of 10 on retain accuracy. We therefore do not claim that SalUn's weights fail to reproduce. What the evidence supports is narrower: its forget accuracy cannot be recovered without its own definition of the forget set, which neither release documents. The other methods reproduce at 71 of 72 admitted checkpoints, 42 of 43 excluded ones and 40 of 40 on Unlearning Comparator.

Table 4: **Checkpoints measured and reproduced, by population and method.** *Reproduced*: reproduces d_f, d_r and d_t within 2 pp with its shipped statistics.

population	family	measured	reproduced	note
MU-Bench CIFAR-100, admitted	all	72	71	–
MU-Bench CIFAR-100, excluded	bad teaching	22	22	–
	random label	21	20	–
	SalUn	30	0	d_f alone: 30 of 30
	all	73	42	–
MU-Bench CIFAR-10	bad teaching	40	40	–
	NegGrad	55	55	–
	random label	55	55	–
	SalUn	55	0	d_f alone: 54 of 55
	all	205	150	–
Unlearning Comparator	retrained reference	10	10	–
	CF-k	10	10	–
	AdvNegGrad	10	10	–
	SalUn	10	2	–
	SSD	10	10	–
	all	50	42	–

Table 5: **On Unlearning Comparator the retrained reference returns to zero, as a method whose statistics were fitted on retained data should.** Shift in retain accuracy under the refit, one checkpoint per deleted class. The mean, its spread across classes and the 90% interval against the ± 1.2 pp margin are from the first refit; the contrast paired with the reference on the same class averages three refits per class. Only 2 of SalUn's 10 checkpoints reproduce, and SSD's mean is set by three checkpoints that fail the release's criterion.

method	n	mean d_r shift (pp)	sd (pp)	90% interval (pp)	verdict	paired vs reference (pp)	p
retrained reference	10	$+0.01$	0.07	$[-0.03, +0.06]$	equivalent	–	–
CF-k	10	$+0.05$	0.07	$[+0.01, +0.10]$	equivalent	$+0.05$	0.21
AdvNegGrad	10	$+1.92$	0.73	$[+1.49, +2.34]$	different	$+1.88$	2×10^{-5}
SalUn	2	$+2.46$	1.04	–	–	–	–
SSD	10	$+11.38$	20.14	–	–	$+11.37$	0.11

C THE REFIT, STATED EXACTLY

The contribution rests on one operation, so we state it as an algorithm rather than a hyperparameter table (Algorithm 1). The weights are loaded strictly from the release's own files, and every batch-normalization layer stays in evaluation mode throughout, so each layer is fitted on inputs that already reflect the finished statistics of the layers before it. The obvious alternative, running the network in training mode and letting the framework average per-batch statistics, depends on the batch partition; the exact procedure does not.

Algorithm 1 Exact refit of batch-normalization statistics at fixed weights

Require: checkpoint with weights θ; normalization layers $\ell_1, \ldots, \ell_L$ in topological order; retained sample S of 8000 images under the producer's training transform
Ensure: statistics $\hat{\phi} = \{(\mu_j, \sigma_j^2)\}_{j=1}^{L}$; θ unchanged
put every layer in evaluation mode
for $j = 1$ **to** L **do**
  $n \leftarrow 0$; $s \leftarrow 0$; $q \leftarrow 0$ {per-channel sums, float64}
  for each batch $B \subseteq S$, in any order and of any size **do**
    $X \leftarrow$ input to ℓ_j on B, computed with $\ell_1, \ldots, \ell_{j-1}$ already refit
    $s \leftarrow s + \sum X$; $q \leftarrow q + \sum X^2$; $n \leftarrow n +$ (values per channel in X)
  end for
  $\mu_j \leftarrow s/n$; $\sigma_j^2 \leftarrow (q - s^2/n)/(n-1)$
  write (μ_j, σ_j^2) into ℓ_j and reset its batch counter
end for
check that the hash of θ is unchanged and the hash of $\hat{\phi}$ has changed

Images, transform and pool. MU-Bench's training script accumulates its statistics under random resized cropping to 224 pixels, random horizontal flipping and the release's normalization; its configuration file disagrees, and the script wins. Scoring uses the release's evaluation transform, a resize and a center crop to 224 pixels, which reproduces the published accuracy of the original model exactly. The refit pool is 8000 retained training images drawn without replacement. Retain accuracy is scored on a disjoint set of 10,000 retained images, so no scored image is ever in the refit pool. One refit draws its pool from a pool seed and its augmentation from an augmentation seed; the census redraws both independently on each of its ten refits. The across-seed spread on the original model is 0.05 pp of test accuracy.

The transform matters and the averaging rule does not. Figure 3 pairs each averaging rule with each image transform. On the original model, refitting under the evaluation transform shifts test accuracy by -2.55 pp with exact statistics and -2.58 pp with a moving average, while the training transform gives -0.08 pp and -0.15 pp. On a bad-teaching checkpoint the two transforms give $+4.13$ pp and $+2.83$ pp. A refit under the wrong transform therefore measures a distribution mismatch on top of the effect, and every number in this paper uses the training transform.

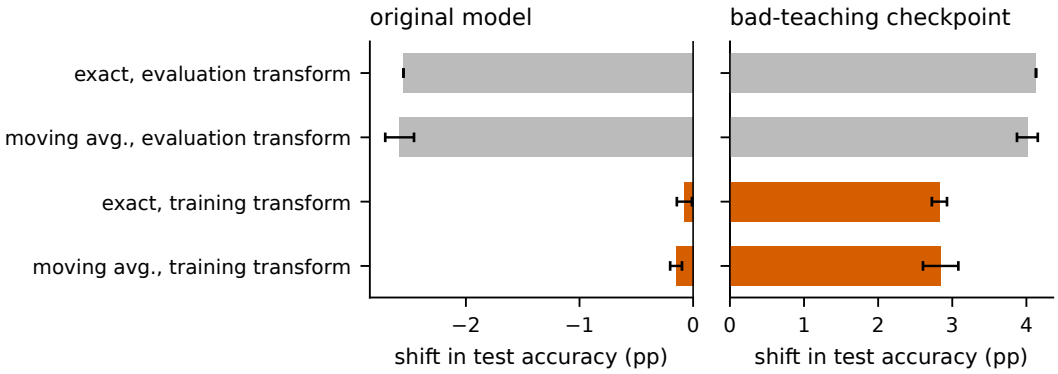

Figure 3: **The refit's image transform moves test accuracy by more than 2 pp; its averaging rule barely matters.** Test-accuracy shift under the four combinations of averaging rule and transform, on MU-Bench's original model (left) and a bad-teaching checkpoint (right); means over three seeds with one standard deviation.

The refit against the only statistics with ground truth. Unlearning Comparator's ten retrained references accumulated their statistics on retained data while their weights trained. We compare each shipped state with our refit of the same checkpoint, channel by channel, in units of that channel's activation standard deviation, against a threshold of 0.10 on the median fixed in advance. The references land at 0.0019. CF-k, AdvNegGrad, SalUn and SSD sit $1.8\times$, $17.5\times$, $19.3\times$ and $47.1\times$ further away, and their order matches the order of their accuracy shifts. The displacement is therefore visible in the statistics themselves, not only in an accuracy.

D THE PER-CHECKPOINT DECISION RULE

Why ten refits. In this appendix we call a count *certified* when it survives the Benjamini–Yekutieli control of Section 3. An exact sign test on r refits can reject at best at $(1/2)^r$, so the multiplicity procedure it can pass depends on r. At eight refits the floor of 0.0039 sits above the Benjamini–Yekutieli threshold for 221 checkpoints, and nothing can be certified under arbitrary dependence. At ten refits the floor of 0.00098 clears it, so the counts hold whatever the dependence between checkpoints of one release. It does not clear Bonferroni, which would need thirteen refits, so we make no familywise claim.

What the count means. The count is a floor, not an estimate of how many checkpoints move: a checkpoint whose true shift sits just above the margin will often have one of its ten refits below it, and then it is not counted. The test certifies the direction of a shift only. The refit itself is deterministic given its draws: re-running an earlier refit of the census from scratch reproduces all 70 checkpoints it covers to within 0.00 pp.

Scatter between checkpoints. Independently trained checkpoints differ before any refit, so the census might measure that scatter. It would need a standard deviation of 0.75 pp to place even the smallest displaced checkpoint past the margin at two standard deviations. That is $6.2\times$ the upper bound from MU-Bench's three original models and $10.3\times$ the spread of Unlearning Comparator's ten independently trained references.

Room to move. A checkpoint whose accuracy is already high might have less room to rise. The forget-accuracy shift correlates with the reproduced forget accuracy at only -0.14 on CIFAR-100 and -0.01 on CIFAR-10, though at -0.70 within random label, which has 8 checkpoints. Table 6 breaks the displaced counts down by release group and method.

The two values we chose. The margin and the pool size are chosen rather than measured, so we swept both (Table 7). Moving the margin from 1.2 to 1.0 pp raises the forget count from 47 to 56, and moving it to 1.5 pp lowers it to 31, monotonically on all three accuracies. At 2.0 pp 23 checkpoints

Table 6: **Displaced checkpoints by release group and method, over ten independent refits.** A checkpoint is displaced when all ten shifts exceed $+1.2$ pp; the pooled counts are certified by Benjamini–Yekutieli at $q = 0.05$. The last column is the method-level verdict on CIFAR-100 forget accuracy; MU-Bench publishes no method-level criterion for its CIFAR-10 groups.

			displaced			family
group	family	n	d_f	d_r	d_t	verdict
CIFAR-100 ResNet-50	bad teaching	18	12	10	10	unresolved
	NegGrad	45	8	6	5	equivalent
	random label	8	0	0	0	equivalent
	total	71	20	16	15	
CIFAR-10 ResNet-50	bad teaching	30	5	4	5	–
	NegGrad	45	10	10	10	–
	random label	45	1	0	0	–
	total	120	16	14	15	
CIFAR-10 ResNet-34	bad teaching	10	3	3	3	–
	NegGrad	10	3	3	3	–
	random label	10	5	3	4	–
	total	30	11	9	10	
pooled		221	47	39	40	

remain, and none is certified: every one sits at the sign test's floor, and Benjamini–Yekutieli needs a tied set of at least 26. Across pools of 4000, 8000 and 16,000 images, the per-checkpoint shift agrees with the 8000-image value to a median of 0.15 and 0.10 pp, and all 71 checkpoints stay within the margin of it. A single-refit count on test accuracy varies more across pools, 28, 22 and 25, which is why the census decides on ten refits rather than one.

Table 7: **The census against its two chosen values.** Top: displaced checkpoints at four margins, at the sign test's floor and as certified. Bottom: agreement of the per-checkpoint forget shift with the 8000-image pool, one refit per pool size, on MU-Bench CIFAR-100.

	d_f		d_r		d_t	
margin (pp)	at floor	certified	at floor	certified	at floor	certified
± 1.0	56	56	48	48	47	47
± 1.2	47	47	39	39	40	40
± 1.5	31	31	28	28	30	30
± 2.0	23	0	21	0	19	0

			$\lvert\Delta\rvert$ against the 8000 pool		
pool size	mean d_f shift (pp)	past margin	median (pp)	worst (pp)	inside margin
4000	$+1.18$	25	0.15	0.60	71 of 71
8000	$+1.19$	26	–	–	–
16000	$+1.19$	25	0.10	0.30	71 of 71

E METHOD-LEVEL INTERVALS, AND WHAT THREE CLUSTERS CAN RESOLVE

MU-Bench releases each method as a grid of three procedure variants, five removal ratios and five seeds, so its checkpoints are not independent draws. Every method-level interval models that structure with cluster-robust variances over the three procedure variants (Table 8, Figure 4a). Across the five estimators we ran, the two equivalent methods keep their verdict in 29 of 30 method, accuracy and estimator combinations; the exception is NegGrad retain accuracy under a crossed hierarchical fit. The dependence is real: on bad teaching the intraclass correlation is 0.15 to 0.23 by variant and 0.34 to 0.37 by ratio.

An interval that contains zero can mean the effect is absent or that the design cannot see it. To separate the two we centre each method's shifts to a true mean of zero, which keeps every cluster and every spread, and plant 0.0 or 3.0 pp (Table 9). The estimator recovers the planted level in all eighteen cases. On NegGrad and random label it also returns the right verdict at both plants, so their equivalences are powered. On bad teaching both plants come back unresolved, because its shift depends strongly on the procedure variant. The method that moves is exactly the one whose average three clusters cannot pin, which is why the paper's unit of decision is the checkpoint.

Table 8: **Two MU-Bench methods are equivalent to zero on every published accuracy; the third cannot be resolved at three clusters.** Mean shift and 90% cluster-robust interval (pp); the last column is MU-Bench's original model, which has nothing removed. The last row is the half-width returned when a true zero is planted into each method's own clusters. G is the number of clusters in the estimator, per accuracy (d_f/d_r/d_t) where they differ.

	shift (pp)			
metric	bad teaching n=18, G=3	NegGrad n=45, G=3/3/15	random label n=8, G=3/5/5	original model n=3 seeds
forget accuracy d_f	+2.47 [−0.12, +5.06]	+0.83 [+0.49, +1.17]	+0.35 [−0.24, +0.94]	+0.21
retain accuracy d_r	+2.23 [+0.08, +4.38]	+0.90 [+0.63, +1.18]	+0.05 [−0.05, +0.15]	−0.05
test accuracy d_t	+1.86 [−0.26, +3.98]	+0.84 [+0.58, +1.10]	+0.10 [+0.03, +0.17]	−0.08
verdict	unresolved	equivalent	equivalent	–
detection floor (pp)				
forget accuracy d_f	±2.59	±0.34	±0.59	–
retain accuracy d_r	±2.15	±0.28	±0.10	–
test accuracy d_t	±2.12	±0.26	±0.07	–

Table 9: **Detection floor by method and accuracy.** Each method's own shifts, centred to zero, with a known shift planted and the clustered estimator re-run.

		planted 0.0 pp			planted 3.0 pp		
family	metric	recovered (pp)	90% interval (pp)	verdict	recovered (pp)	90% interval (pp)	verdict
bad teaching	d_f	0.00	[−2.59, +2.59]	unresolved	+3.00	[+0.41, +5.59]	unresolved
	d_r	0.00	[−2.15, +2.15]	unresolved	+3.00	[+0.85, +5.15]	unresolved
	d_t	0.00	[−2.12, +2.12]	unresolved	+3.00	[+0.88, +5.12]	unresolved
NegGrad	d_f	0.00	[−0.34, +0.34]	equivalent	+3.00	[+2.66, +3.34]	different
	d_r	0.00	[−0.28, +0.28]	equivalent	+3.00	[+2.72, +3.28]	different
	d_t	0.00	[−0.26, +0.26]	equivalent	+3.00	[+2.74, +3.26]	different
random label	d_f	0.00	[−0.59, +0.59]	equivalent	+3.00	[+2.41, +3.59]	different
	d_r	0.00	[−0.10, +0.10]	equivalent	+3.00	[+2.90, +3.10]	different
	d_t	0.00	[−0.07, +0.07]	equivalent	+3.00	[+2.93, +3.07]	different

F PROVENANCE OF THE REFIT POOL, AND THE WEIGHT-DISTANCE LADDER

Design of the three fits. In each of MU-Bench's three groups, the 20 checkpoints are the ten most and ten least displaced in that group's census, selected by rank from the census alone. All three fits use 8000 images, and the weights, estimator and transform are fixed (Section 4.2). With n_{del} removed training images out of 50,000, $k = \mathrm{round}(8000\, n_{\mathrm{del}}/50000)$. The retained-only fit and the fit with removed images at their training-set share use a common core of $8000 - k$ retained images and differ only in whether their last k images are retained or removed. The fit with every removed image holds all n_{del} removed images plus $8000 - n_{\mathrm{del}}$ retained ones; at the largest removal ratio that is 5000 removed images of 8000. Each CIFAR-100 checkpoint gets six independent draws and each

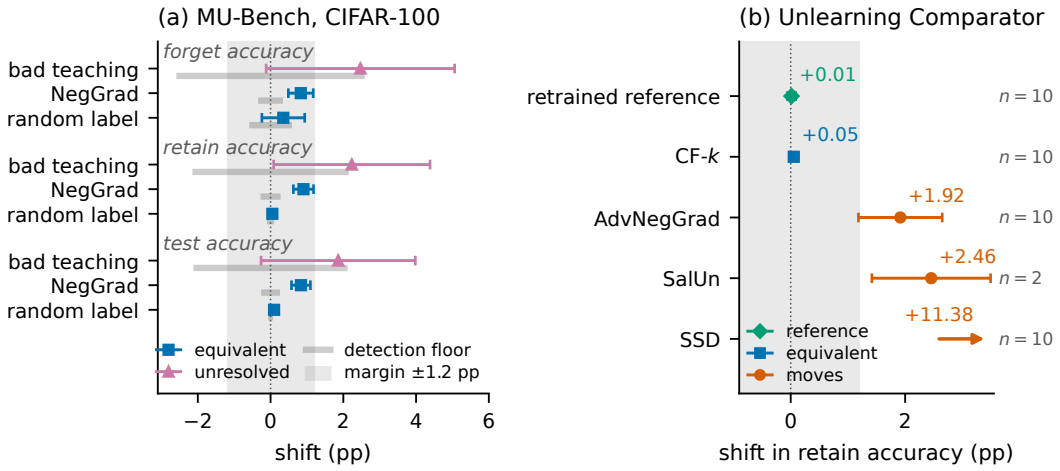

Figure 4: **At identical architecture, data and refit, one method's numbers return to zero while another's move past the margin.** (a) MU-Bench CIFAR-100: method-level 90% intervals against the ± 1.2 pp margin, with each method's detection floor as a grey bar. (b) Unlearning Comparator: mean shift in retain accuracy per method with one standard deviation; SSD is off the scale because of three checkpoints that fail the release's criterion.

CIFAR-10 checkpoint thirteen. On CIFAR-100 the retained-only fit reproduces the census's own first-draw forget accuracy exactly on all 20, and all 120 checkpoint-draws reproduce their published numbers before any contrast is read. On CIFAR-10, a fit that repeats the census's first draw image for image reproduces its forget accuracy exactly on 40 of 40 checkpoints, and all 520 checkpoint-draws reproduce their published numbers. As a control, swapping the last k retained images for k other retained images moves forget accuracy by $+0.12$ pp on ResNet-50 and by $+0.02$ pp on ResNet-34 at the first draw.

What the three fits return. On CIFAR-100 the contrast against the retained-only fit is positive on 5 of 6 draws for the fit with removed images at their training-set share and on 4 of 6 for the fit with every removed image. At six draws the exact two-sided floor is 0.031, so nothing is certified in either direction; what is measured is the size, and it is small (Table 2). A null on the published numbers can mean the statistics changed and the numbers could not show it, or that the statistics did not change. Panel (a) of Table 2 decides between the two on CIFAR-100 and CIFAR-10 ResNet-34: the distances between fits with and without removed images stay inside the band at which our refit reproduces a retrained reference, so there is no change for the numbers to hide. On quantization scales the opposite holds (Appendix I). Table 10 splits the CIFAR-100 checkpoints by their mean shift. The shipped statistics of the ten with the largest shift sit farther from the retained-only fit than those of the ten with the smallest, while removed images at their training-set share move both alike.

CIFAR-10 ResNet-50, where removed images do reach forget accuracy. At thirteen draws the exact two-sided floor is 0.00024. On this group the pooled contrast against the retained-only fit is positive on all 13 draws for the fit with removed images at their training-set share, at $+0.25$ pp, and is $+1.50$ pp for the fit with every removed image (Table 11). Four checkpoints are positive on all 13 draws, the four random-label checkpoints in the selection, and two of them reach the 1.2 pp margin. Under the decision rule fixed in advance, the null that removed images in the fit do not matter fails on this group at the training-set share and holds on CIFAR-10 ResNet-34. That the responders are exactly the random-label checkpoints is a pattern noticed afterwards, not a prediction. Three of the four sit among the ten least displaced, and the stratum means run against the displacement: $+0.12$ pp for the ten most displaced and $+0.38$ pp for the ten least. At the median, the fit with removed images at their training-set share sits as far from the retained-only fit as the swap control does, yet 2 of 20 lie beyond the refit-noise level. The median distance therefore does not flag the checkpoints whose forget accuracy responds. Closing this group would take the contrast on its remaining random-label checkpoints and an account of why only those respond.

Table 10: **CIFAR-100 provenance test, split by mean forget-accuracy shift.** The 20 CIFAR-100 checkpoints of Table 2, as the ten with the largest and the ten with the smallest mean shift. Distances as in Section 3, with the count beyond the refit-noise level.

(a) distance from the retained-only fit

			by mean shift	
fit	median	beyond noise level	10 largest	10 smallest
shipped statistics	0.0054	20 of 20	0.0081	0.0029
removed images, training-set share	0.0006	0 of 20	0.0006	0.0006
every removed image	0.0010	1 of 20	–	–
refit-noise level	0.0019	–	–	–

(b) forget accuracy d_f, pooled over 6 draws

			by mean shift	
fit	mean (pp)	past 0.5 pp	10 largest (pp)	10 smallest (pp)
removed images, training-set share	+0.02	0 of 20	+0.01	+0.02
every removed image	+0.06	0 of 20	+0.10	+0.01

Table 11: **On CIFAR-10 ResNet-50, removed images move forget accuracy on the four random-label checkpoints, not on the four bad-teaching checkpoints the refit displaces most.** Every checkpoint in this group's removed-image test, ordered by its census shift, the ten-refit mean forget-accuracy shift (pp). *Stratum*: among the census's ten most or ten least displaced. *Training-set share* and *every removed image*: change in forget accuracy of the fit with removed images at their training-set share and of the fit with every removed image, against the retained-only fit, mean over 13 draws (pp). *Draws positive*: draws on which the change at the training-set share is above zero. *Ratio*: the removal ratio; *variant*: MU-Bench's procedure variant.

method	variant	ratio	seed	stratum	census shift	training-set share	draws positive	every removed image
bad teaching	second-order	10%	87	most	+19.94	−0.01	4 of 13	−0.95
bad teaching	second-order	10%	42	most	+18.87	0.00	4 of 13	−0.50
bad teaching	second-order	8%	87	most	+13.64	−0.06	2 of 13	−0.30
bad teaching	second-order	6%	42	most	+11.28	+0.07	10 of 13	−0.16
NegGrad	second-order	10%	21	most	+4.30	+0.08	10 of 13	+0.21
NegGrad	second-order	8%	100	most	+3.14	+0.09	8 of 13	+0.18
NegGrad	second-order	8%	87	most	+2.79	+0.08	8 of 13	+0.18
random label	second-order	4%	42	most	+2.57	+0.96	13 of 13	+5.71
NegGrad	second-order	10%	100	most	+2.56	+0.02	7 of 13	+0.09
NegGrad	second-order	6%	100	most	+2.11	+0.01	7 of 13	−0.01
NegGrad	second-order	4%	21	least	−0.53	−0.02	4 of 13	+0.09
random label	second-order	6%	42	least	−0.56	+1.20	13 of 13	+7.10
bad teaching	base	8%	87	least	−0.64	−0.02	5 of 13	+0.21
random label	curriculum	10%	42	least	−0.67	+1.09	13 of 13	+7.79
bad teaching	curriculum	4%	42	least	−0.73	+0.03	8 of 13	+0.81
bad teaching	base	2%	87	least	−0.75	+0.15	8 of 13	+0.29
bad teaching	base	2%	42	least	−0.77	−0.01	6 of 13	+0.08
random label	second-order	10%	42	least	−0.83	+1.46	13 of 13	+9.19
bad teaching	second-order	2%	42	least	−1.06	+0.01	6 of 13	+0.03
bad teaching	curriculum	2%	87	least	−1.59	−0.05	5 of 13	+0.02

The ladder. On the SVHN population's own original models, each of three methods (SalUn, fine-tuning on retained data, and gradient ascent) runs along six steps of optimization strength, and each step runs twice: at the implementation's default, which updates the statistics during unlearning, and with only that update suppressed. We call the suppressed run the default run's *twin*, and the forget-accuracy difference between them the *twin gap*. That gives 180 runs over five seeds, and inside every pair the trained weights are bitwise identical. A method that updates its statistics over removed data is then compared with one that does not at matched median relative weight distance from the original model, within $|\ln(\text{ratio})| \leq 0.2$, a rule fixed before any run was scored (Table 12).

Only one matched pair keeps its frozen twin usable on all five seeds, and there both methods sit at the forget metric's own resolution: the largest single-seed twin gap is 0.53 pp against a detection floor of 12.93 pp. The two matched pairs that do resolve, at +13.04 and +51.37 pp, keep their frozen twins usable on 3 and 0 of 5 seeds. Using every step instead, SalUn's twin gap sits +33.80 pp above what the fine-tuning curve of twin gap against weight distance predicts at the same distance. At the unmatched default operating point the same comparison gives +65.31 pp with the same sign on all five seeds, so the instrument can see an effect of this kind. Most suppressed runs are not usable models, so the ladder bears on how the statistics' contribution relates to weight movement, not on shippable checkpoints or on whose data the statistics saw.

Table 12: **Read over every ladder step, SalUn's twin gap exceeds fine-tuning's at equal weight distance. The distance-matched reading stays unresolved: its one pair with usable frozen twins sits at the metric's resolution, and the pairs that resolve keep usable twins on 3 and 0 of 5 seeds.** Top: pairs of ladder steps matched on weight distance alone. Bottom: the reading over every step, and checks on the instrument.

steps (SalUn / fine-tune)	| ln ratio|	frozen twin usable	twin gap (pp)	sd (pp)
1 / 0	0.117	5 of 5	+0.07	0.21
2 / 1	0.055	3 of 5	+13.04	28.07
5 / 3	0.092	0 of 5	+51.37	25.71

quantity	value
detection floor (pp)	12.93
gap above other curve (pp)	+33.80
interval, 5 seeds (pp)	[+21.28, +46.32]
default operating point (pp)	+65.31
its sd (pp)	10.57
seeds agreeing in sign	5 of 5
twins, identical weights	90 of 90
default steps, reproduction (pp)	0.00

G EVERY CANDIDATE FLIPPED VERDICT

Table 13 lists all 16 candidate flipped verdicts and reconciles the totals of Table 3. Figure 5 draws each as an arrow from its shipped to its refit criterion value. The CIFAR-100 flipped verdicts start and end near the bound, within twice the baseline of it; the two CIFAR-10 ones travel deep into the admitted region.

Stability across refits. A flipped verdict declared on one refit could depend on that refit's draw, so we re-decided every flipped verdict with ten refits on each of them, under the published-number and recomputed-number rules. The two CIFAR-10 flipped verdicts hold on all ten under either rule, and that group's count per refit does not move. On CIFAR-100, of the six flipped verdicts that hold on the published number and have ten refits, four hold on all ten, one on nine and one on six, and the count per refit ranges from 6 to 9. Eleven verdict-and-rule pairs are unanimous, each at the exact floor of 0.00098, and all clear the Benjamini–Yekutieli cutoff of 0.017. Four verdicts that flip on some refits but not on the first are reported in the result files and counted nowhere. With every shift set to zero, the decision code flips no verdict.

H SELF-TRAINED POPULATIONS

We found no third release that publishes a per-checkpoint table an outsider can reproduce, so we built two populations. Both are self-trained, reported as such, and pooled with neither release. Both use a public unlearning implementation at a fixed version and at its own defaults (Fan et al., 2024), with one disclosed one-line import repair. Each of five seeds has its own original model and its own retrained reference, and every method runs twice: at the implementation's default, which updates

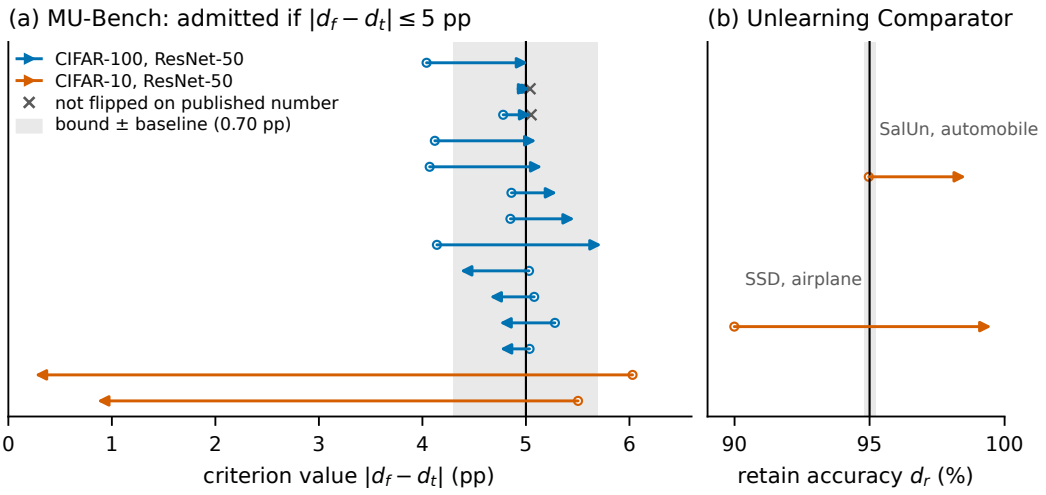

Figure 5: **Flipped verdicts on MU-Bench CIFAR-100 stay near the bound; the decisive ones travel far.** Each arrow runs from a checkpoint's shipped criterion value to its refit value. (a) MU-Bench, with the band of ± the CIFAR-100 baseline around the bound; an × marks each verdict that does not flip on the published number. (b) Unlearning Comparator's two flipped verdicts on its retain-accuracy bar.

the statistics during unlearning, and with only that update suppressed. Each population's margin is set the way MU-Bench's is, as the median standard deviation across seeds of its own accuracy.

SVHN, VGG-16-BN. Table 14 gives every method. SalUn is the implementation's masked random relabelling, and the only method here whose updates mix removed and retained data. On its four checkpoints that pass both the gap criterion and a utility floor referenced to their seed's retrained model, the refit moves forget accuracy $+0.47$ pp on average, and 2 of 4 clear the 0.32 pp margin. The three controls in the same run return to zero: fine-tuning on retained data -0.01 pp, the original model -0.01 pp and the retrained reference $+0.02$ pp. These magnitudes are an order below the released populations', so this population shows reach, not a comparable effect size. It also shows something about the criterion: all 40 checkpoints pass $|d_f - d_t| \leq 5$ pp, and 19 of them sit far below their own retrained reference, since a model at chance passes a gap test trivially. Suppressing the statistics update leaves weights bitwise identical and changes test accuracy by -9.50 to -75.81 pp, so shipped statistics carry behavior the weights do not determine. Fine-tuning on retained data pays that cost too, so it is the mismatch between statistics and moved weights that costs accuracy, not the removed data.

CIFAR-100, VGG-16-BN, built near the criterion. Optimization strengths were chosen by a rule fixed before any original model was trained: from a first ladder of settings, pick the one whose gap $|d_f - d_t|$ lands closest to 5 pp. The five original models reach test accuracies of 72.00% to 72.42%, inside a published range read before they were scored. The refit instrument is bit-identical to the SVHN one, and over 990 checkpoint-draws no weight tensor moves and no refit repeats. Against the implementation's own evaluation code, the worst disagreement is 0.02 pp on forget and 0.01 pp on test accuracy over 100 checkpoints; one checkpoint whose retain probe disagrees by 1.02 pp is excluded. Table 15 gives the result. Five verdicts do flip in the same runs, all on frozen-statistics twins, which is our intervention rather than anything the implementation ships, so they enter no count.

I A SECOND FITTED STATE: QUANTIZATION SCALES

Static post-training quantization maps each activation to 8-bit integers through one scale per tensor, fitted by passing calibration data through the network. We attach the quantizer outside the model, so trained weights and normalization statistics are untouched and the calibration set is the only

Table 13: **All 16 candidate flipped verdicts.** Criterion values are our recomputation before and after the refit: $|d_f - d_t|$ in pp for MU-Bench, and d_r in percent (marked %) for Unlearning Comparator. *Published*: the verdict also flips on the release's printed number. *Past bound*: distance beyond the bound after the refit. *Above baseline*: past the bound by more than the baseline. *Stable*: refits of ten on which the verdict flips, for checkpoints with ten refits.

family	variant or class	removal ratio (%)	seed	direction	criterion value (pp) before → after	holds on published	past bound (pp)	above baseline	stable (of 10)
MU-Bench, CIFAR-100 ResNet-50									
bad teaching	base	2	42	out	4.14 → 5.73	yes	0.73	marginal	10 of 10
bad teaching	second-order	2	21	out	4.85 → 5.47	yes	0.47	no	10 of 10
bad teaching	curriculum	4	87	out	4.86 → 5.30	yes	0.30	no	10 of 10
bad teaching	base	6	42	out	4.07 → 5.16	yes	0.16	no	9 of 10
bad teaching	base	2	87	out	4.12 → 5.10	yes	0.10	no	10 of 10
random label	curriculum	4	87	out	4.78 → 5.05	no	0.05	no	2 of 10
random label	second-order	4	87	out	4.96 → 5.04	no	0.04	no	9 of 10
bad teaching	second-order	8	100	out	4.04 → 5.02	yes	0.02	no	6 of 10
random label	base	8	87	in	5.03 → 4.37	yes	0.63	no	–
bad teaching	base	4	87	in	5.08 → 4.65	yes	0.35	no	–
bad teaching	second-order	8	13	in	5.28 → 4.74	yes	0.26	no	–
random label	base	8	42	in	5.04 → 4.75	yes	0.25	no	–
MU-Bench, CIFAR-10 ResNet-50									
bad teaching	second-order	10	87	in	6.03 → 0.26	yes	4.74	yes	10 of 10
bad teaching	second-order	6	42	in	5.50 → 0.86	yes	4.14	yes	10 of 10
Unlearning Comparator, CIFAR-10 ResNet-18									
SSD	airplane	–	–	fail→pass	89.99% → 99.51%	–	4.51	yes	–
SalUn	automobile	–	–	fail→pass	94.97% → 98.56%	–	3.56	yes	–

count	MU-Bench: CIFAR-100 ResNet-50	MU-Bench: CIFAR-10 ResNet-50	MU-Bench: CIFAR-10 ResNet-34	Unlearning Comparator	total
candidate flipped verdicts	12	2	0	2	16
hold on published number	10	2	0	–	12
decisively above baseline	0	2	0	2	4
marginally above baseline	1	0	0	0	1
published and decisively above	0	2	0	–	2

Table 14: **On SVHN the displacement reappears on the one method that mixes removed and retained data, and the controls return to zero.** Forty checkpoints at ten refits each. *Usable*: passes the gap criterion and a utility floor. *Frozen* rows have weights bitwise identical to their default sibling and differ only in the shipped statistics.

method	statistics	removed data exposure	n	usable	shipped d_t (%)	mean d_f shift (pp): all	mean d_f shift (pp): usable	mean d_f shift (pp): displaced
original model	–	–	5	5	95.70	−0.01	−0.01	0
omission retrain	default	none	5	5	95.48	+0.02	+0.02	0
retain-only fine-tune	default	none	5	5	95.74	−0.01	−0.01	0
retain-only fine-tune	frozen	none	5	0	86.24	+11.12	–	0
gradient ascent	default	pure forget	5	1	24.48	0.00	0.00	0
gradient ascent	frozen	pure forget	5	1	24.47	0.00	+0.02	0
SalUn	default	mixed	5	4	88.82	+3.18	+0.47	2
SalUn	frozen	mixed	5	0	13.01	+79.61	–	0

thing that changes. The ResNet-50 checkpoints carry 52 quantized sites and the Swin checkpoints 149, calibrated by the absolute-maximum rule on 512 images. That operating point was fixed on a checkpoint with nothing removed. Quantization moves forget accuracy by -1.65 pp on ResNet-50 and -0.21 pp on Swin, so the quantized models remain usable.

Two contrasts are measured over thirteen independent draws (Table 16). The displacement contrast recomputes each published number under a retained-only calibration set and compares it with the number the checkpoint produces as shipped, against the 1.2 pp margin. The exact contrast compares calibration sets that share a retained core and differ only in whether k of their images are removed

Table 15: **Built near its criterion, this population shows the displacement at released scale and no flipped verdict.** Top: the ten-refit census on the 45 checkpoints the producer's default policy ships, against the population's own margins. Bottom: what a flipped verdict would have needed. *Moved away*: the refit widens the admission gap. *Above bound*: the gap already exceeds 5 pp, so only a narrowing could flip the verdict.

metric	n	margin (pp)	displaced	median sd (pp)
forget accuracy d_f	45	1.87	20	0.10
retain accuracy d_r	45	1.29	20	0.10
test accuracy d_t	45	0.80	22	0.07

quantity	value
baseline (pp)	0.09
verdicts flipped, default policy	0 of 45
moved away from the bound	33 of 45
above the bound	30 of 45
smallest convertible gap shift (pp)	2.36
largest observed gap shift (pp)	3.00
nearest miss (pp)	0.20

or retained, so under the null the sign of the difference is a fair coin. Thirteen draws on one side then give $p = 0.00024$, below the familywise threshold of 0.00042 for 118 checkpoints. Neither contrast moves any checkpoint. The scales themselves do change: a median 12.1% of ResNet-50 scales and 11.2% of Swin scales differ with the removed images in or out. Here the null is about propagation to the published numbers, not about the state. The absolute-maximum rule reads an extreme order statistic, which one image can set, while averaging rules dilute each image's influence as the calibration set grows; neither route carried enough change to reach an accuracy.

Table 16: **The same census on quantization scales returns zero on both architectures, including the one with no normalization statistics.** Thirteen independent draws per checkpoint. The same-sign rows count checkpoints whose thirteen draws share one sign, for calibration sets that differ in whether they contain removed images and, as a control, for two sets of retained images. Detection floors are the planted shift recovered on at least 90% of a group's checkpoints. The all-removed rows calibrate entirely on removed images, which bounds the channel rather than describing a deployment.

quantity	ResNet-50 batch norm	swin-base layer norm
reproduced checkpoints	71	47
quantization sites per checkpoint	52	149
displaced past margin, d_f	0	0
displaced past margin, d_r	0	0
displaced past margin, d_t	0	0
same sign, removed vs retained	0	0
same sign, retained vs retained	0	0
largest |mean| d_f shift (pp)	0.29	0.17
original model, |mean| (pp)	0.05	0.04
detection floor, sign test (pp)	1.00	0.50
detection floor, census (pp)	2.00	2.00
share of scales that move	0.121	0.112
all-removed, past margin	3 of 71	0 of 47
all-removed, largest shift (pp)	6.14	0.27

J FORGETTING QUALITY

MU-Bench cannot answer. The standard membership attack is at chance on these checkpoints: its cross-validated accuracy has a median of 51.8% against a readability floor of 55%, because the models barely overfit. The privacy number the field publishes still moves under the refit, by -0.79 pp pooled, but no count can be certified. A drift readout on the same checkpoints is voided by its own control: the original model, with nothing removed, drifts 0.0642 along the same axis against 0.0191 for the unlearned checkpoints, $3.36\times$ further.

Unlearning Comparator answers for one method. This release ships a retrained oracle for every deleted class, so a checkpoint can be located between the original model and its oracle. Let r be the mean probability a model assigns to the deleted class on that class's held-out test images. A checkpoint's position is

$$\lambda = \frac{r_{\text{ckpt}} - r_{\text{oracle}}}{r_{\text{orig}} - r_{\text{oracle}}},$$

which is 1 at the original model and 0 at the oracle. The shift $\Delta\lambda$ is λ after the refit minus λ as shipped, the latter read against the endpoints as shipped. For the refit checkpoint we place the endpoints two ways, as shipped or with both refit under the same convention, and report the second. Over three independent refits and under both ways of placing the endpoints, SSD moves $+0.0167$ of that distance back toward the original model on 10 of 10 classes (Table 17). The effect is $+0.0215$ on the 7 classes that clear the release's retain bar and $+0.0054$ on the 3 that fail it, a split declared after the first refit and confirmed by the other two. The other methods stay at noise. The obvious rival, that SSD simply moved its weights furthest, runs the wrong way: SalUn moves its weights $2.4\times$ further and drifts $3\times$ less, and within SSD the classes whose weights moved furthest drift least (Spearman $\rho = -0.79$, permutation $p = 0.0085$). The rule we fixed in advance for this rival nevertheless scores it as mixed rather than refuted, and which property of SSD's procedure produces the effect remains open.

Table 17: **One method of four moves on a forgetting-quality readout measured against a real retrained oracle.** $\Delta\lambda$: position shift toward the original model under the refit, as a fraction of the distance from the original model to the retrained oracle, by method. *Weight moved*: median relative L_2 distance of the trained weights from the original model.

						within family	
family	n	$\Delta\lambda$	positive on all draws	past 0.02	weight moved (rel. L_2)	ρ	perm. p
SSD	10	$+0.0167$	10	5	0.245	-0.79	0.0085
SalUn	2	$+0.0058$	2	0	0.577	–	–
AdvNegGrad	10	0.0000	4	0	0.182	-0.18	0.63
CF-k	10	-0.0001	2	0	0.093	-0.52	0.13
SSD, classes clearing retain bar	7	$+0.0215$	7	–	–	–	–
SSD, classes failing it	3	$+0.0054$	–	–	–	–	–
SSD, sd across draws		0.0001					
SSD, sd across classes		0.0119					

K CHECKPOINT-READABLE SCREENS

An auditor would like to know from the file alone whether a checkpoint is affected. Two quantities readable without data fail within every method: the distance between a checkpoint's shipped statistics and the original model's, and the batch count each layer records. The third, the relative distance of the trained weights from the original model, was tested under a rule fixed before any correlation was computed: uphold if the within-method rank correlation is significantly positive in most method groups, reject if in none, and report a mixed result otherwise. The result is mixed (Table 18): significantly positive in 4 of 9 groups, all on CIFAR-10 and up to $\rho = +0.77$, and in none of CIFAR-100's three. Pooled over all 221 checkpoints it is $+0.09$ at $p = 0.17$, which decides nothing, since pooling methods with different means mixes between-method and within-method variation. Weight

distance is a triage signal within some CIFAR-10 groups and none on CIFAR-100; directional or layer-restricted weight measures are untested.

Reading each release's own training code for whether a method updates its statistics over batches that contain removed data gives a method-level prediction made blind to the shifts (Table 19). Methods whose code updates the statistics on batches made only of removed data average $+1.90$ pp, against $+0.03$ pp for CF-k and the retrained reference, which pass no removed data through them in training mode, a separation of 1.86 pp. Each release contributes its headline accuracy (forget on MU-Bench, retain on Unlearning Comparator), and two of the seven classifications are inferred rather than read. The ordering test returns $p = 0.056$, the smallest value it can reach with three and four methods. On the SVHN population, which the reading had never seen, the methods it classes as unexposed return -0.01 pp and $+0.02$ pp, as predicted.

Table 18: **Weight distance tracks displacement within some CIFAR-10 groups and none on CIFAR-100.** Spearman correlation between a checkpoint's relative weight distance from its original model and its mean forget shift over ten refits, with a within-method permutation test.

group	family	n	mean weight moved (rel. L_2)	mean d_f shift (pp)	Spearman ρ	perm. p
CIFAR-10 ResNet-34	bad teaching	10	0.029	+0.68	−0.15	0.68
	NegGrad	10	0.006	+0.74	+0.77	0.014
	random label	10	0.840	+1.69	−0.02	0.97
CIFAR-10 ResNet-50	bad teaching	30	0.057	+2.22	+0.53	0.0027
	NegGrad	45	0.004	+0.71	+0.43	0.0033
	random label	45	0.260	+0.39	+0.35	0.021
CIFAR-100 ResNet-50	bad teaching	18	0.016	+2.48	−0.34	0.16
	NegGrad	45	0.004	+0.74	+0.16	0.3
	random label	8	0.038	+0.36	+0.60	0.13
pooled		221			+0.09	0.17

Table 19: **Which methods move is predicted by reading the releases' own code.** Whether a method's released unlearning loop updates normalization statistics over batches containing removed data (high), rarely (low) or never (zero), and whether that was read from the executed code path or from the method's definition. The shift is each method's unweighted mean, over its checkpoints, of the release's primary metric; Table 8 gives cluster-robust estimates, which weight checkpoints differently.

release	family	exposure	source	metric	n	mean shift (pp)	displaced
MU-Bench CIFAR-100	bad teaching	HIGH	read	d_f	18	+2.48	12
	NegGrad	HIGH	read	d_f	45	+0.74	8
	random label	LOW	read	d_f	8	+0.36	0
Unlearning Comparator	SalUn	HIGH	read	d_r	2	+2.46	–
	AdvNegGrad	HIGH	inferred	d_r	10	+1.92	–
	CF-k	ZERO	inferred	d_r	10	+0.05	–
	retrained reference	ZERO	read	d_r	10	+0.01	–

quantity	value
mean shift, HIGH families (pp)	+1.90
mean shift, ZERO families (pp)	+0.03
separation (pp)	1.86
margin (pp)	1.2
ordering test, exact p	0.056
smallest attainable p	0.056

L THE LITERATURE RARELY RECORDS THE CONVENTION

Sample. Every unlearning paper we hold as a PDF: a corpus query for recent unlearning evaluation work at ICLR, ICML and NeurIPS 2025–26, plus the prior work this paper positions against. That is 52 unique papers. A paper is included if it uses the word "unlearn" at least fifteen times, a rule fixed in advance; 42 pass. The population is defined by that rule and is not an exhaustive survey of the field.

Detector. Each PDF is converted to text and flattened, removing hyphenation and all whitespace, so that spellings such as "batch norm", "BatchNorm" and a hyphen broken across a line all collapse before matching. We then search for batch-normalization terms, running-mean and running-variance terms, and terms for population or normalization-layer statistics. The concurrent paper about this channel returns 13 hits (Jawandhia et al., 2026), and the detector returns zero hits on an ablated corpus with those terms removed and on a decoy term list, against 417 hits on the unaltered corpus.

Result. Of the 42 included papers, 19 run batch-normalized convolutional models against a model retrained on the retained data. Four of those 19 use any term for the normalization statistics. Without the concurrent paper it is 3 of 18, at 2 to 3 hits each: a method that needs no retained data (Cheng et al., 2026), one that needs no forget set (Newatia et al., 2026) and a tamper-resistance method (Siddiqui et al., 2025). The count measures what papers document, not what their authors did: a paper that never names the convention may still have applied one consistently.

M A SECOND FORGETTING SCORE

MU-Bench publishes a zero-retrain-forgetting score beside its accuracies. It compares a checkpoint's outputs on the removed data with those of a randomly initialized model, so it reads the same statistics through a different summary. We ran the census on it with three refits per checkpoint against a margin of 0.0028 taken from the release's own spread. The recomputed accuracy shifts reproduce the main census on 71 of 71 checkpoints, 70 of 71 reproduce their published score, and the original model moves -0.0007, inside the margin. Table 20 gives the result: 21 of 72 checkpoints move past the margin, 18 up and 3 down, and bad teaching's mean shift of $+0.0115$ is about four times the margin. The count rests on three refits, whose exact test cannot certify, so it corroborates the method pattern without a certified count.

Table 20: **An independent forgetting score separates the same methods more sharply than accuracy does.** Zero-retrain-forgetting shift under the refit, MU-Bench CIFAR-100, three refits per checkpoint, against the release's margin M and a wider sensitivity margin M_{s}, over the 72 checkpoints the criterion admits, including the one whose accuracies do not reproduce.

				displaced	
family	n	mean ZRF shift	median |ZRF shift|	M	M_{s}
bad teaching	18	$+0.0115$	0.0082	18	14
NegGrad	45	-0.0012	0.0014	3	2
random label	9	-0.0009	0.0005	0	0
all families	72			21	16
original model	1	-0.0007		0	0

N EXTENDED RELATED WORK

What the evaluation literature varies. Unlearning evaluations read accuracy over the removed and retained data and often a membership-inference score (Graves et al., 2021; Kurmanji et al., 2023; Jia et al., 2023; Carlini et al., 2022); work on generative models applies criteria of the same shape to generation scores (Maini et al., 2024; Li et al., 2024; Shi et al., 2025). Hayes et al. (2025) and Goel et al. (2022) show that these audits depend on the attack chosen. Other work replaces the

statistic (Seo et al., 2025; Shi et al., 2026; Kim et al., 2026), moves the intervention to inference time (Chowdhury et al., 2026), enumerates the controls an audit claim needs (Sommer et al., 2022; Sarmasarkar et al., 2026; Song et al., 2026), or models the game between auditor and operator (Lin et al., 2026). Li et al. (2026) calibrate the audit metric against a retrained policy. Each of these varies the metric, the attack or the setting. None varies the fitted state the metric is computed through, and a calibrated metric read from two models with mismatched statistics inherits the same term. Zhao et al. (2024) trace how hard a removal is to the forget set's entanglement with retained data and to how memorized it is, properties of the data and the weights; the variation we report sits in the statistics at fixed weights. Thudi et al. (2022a) characterize what makes an approximate removal verifiable, and Kim et al. (2026) argue that logit-level readouts overstate forgetting; our second forgetting score (Appendix M) changes the readout while holding the statistics fixed, and the displacement survives.

Unlearned models can be made to remember. Quantizing an unlearned model can restore forgotten content (Zhang et al., 2025; Khan & Sohel, 2026); normalization statistics amplify memorization during training (Doan et al., 2026); and fine-tuning on retained data can recover behavior an unlearned model was said to have lost (Siddiqui et al., 2025). Dehghani et al. (2026) recover forgotten classes from the unlearned model alone, by updating its classifier head on synthetic probes. These results transform a fixed model from outside or update its weights, and they measure the return of forgotten content. Here the checkpoint's own shipped statistics are the variable. On Unlearning Comparator what the refit restores is utility on retained and test data, and forget accuracy stays at 0.00% wherever it does. Their quantization is data-free and applied to weights, whereas the scales of Appendix I are fitted on data and applied to activations. Kalani et al. (2026) are closest. They formalize the weight-preserving recalibration as a fixed-point operator, prove that the gap it opens is attributable to the running statistics rather than the weights, and measure it on nine self-trained methods, where a normalization-free control removes it. We adopt the operation. Their operator identity says what a gap is attributable to. It does not say what the published record does under the operation, or whether the movement is removed data in the fit, and those are the questions this paper answers. Their large reversals and our small shifts need not conflict: on checkpoints that fail their own criterion the refit also restores tens of percentage points of retain accuracy, while our counts are read on checkpoints that reproduce their published numbers and, on MU-Bench CIFAR-100, pass its criterion.

Why the statistics matter. Normalization parameters are not incidental to what a network computes. Training only the normalization layers of an otherwise random network already reaches non-trivial accuracy (Frankle et al., 2021; Santurkar et al., 2018), so a state that no release records and no audit matches carries part of the function being audited.

Variation in benchmark numbers. Bouthillier et al. (2021) bound how far seed and data resampling move a benchmark result, and the reproducibility literature documents how often a reported number fails to survive re-implementation (Pineau et al., 2021; Raff, 2019; Lucic et al., 2018). Those sources are random and average out over trials; a mismatch between shipped statistics and weights is systematic and does not. Reproducibility tooling for unlearning standardizes the pipeline rather than the state a checkpoint ships (D'Angelo et al., 2025). Two recent audits follow the discipline we use, admitting no claim that does not exceed a measured null on a frozen model (Xu et al., 2026; Zhu, 2026).

O Dates relative to concurrent work

The recalibration operation is not new here, and these dates only place our measurements beside two concurrent papers. The released code of Jawandhia et al. (2026) added a recalibration diagnostic on 24 June 2026; the paper's initial arXiv version, of 2 June 2026, does not discuss it. Our refit, margin and decision rules were fixed in writing on 29 August 2026, and the census measurements began on 30 August 2026. The quantization channel's operating point and detection floors were fixed on 6 September 2026. Kalani et al. (2026) was posted on 8 September 2026, after these dates and after the census this paper reports.