WADE: WASSERSTEIN DISTRIBUTIONAL ALIGNMENT FOR KNOWLEDGE ERASURE IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Machine unlearning for large language models aims to remove the influence of specified forget data while preserving retained knowledge. Retraining on the retained set is the gold standard, but is prohibitively expensive. Existing approximate methods typically optimize proxy forget losses or steer forget activations toward a single target vector, which can suppress forgotten answers without matching the distribution of forget-query behavior. We propose WADE, a closed-form activation-editing method that casts LLM unlearning as distributional alignment in activation space. WADE edits selected MLP down-projection matrices so that the mean and covariance of forget-query activations match a target distribution induced by unknown-entity prompts, under a second-order Wasserstein objective with a retain-key regularizer. A Bures–Procrustes reduction yields a non-iterative, ridge-regularized weight update that provably does not increase the surrogate objective. On TOFU across three LLM families, WADE attains the lowest KS distance to the retrain-only reference among the evaluated baselines at their published settings; under per-method hyperparameter sweeps it provides the low-KS end of the alignment–utility frontier, while NPO attains higher utility and K-FADE reaches a neighboring point at much higher cost. Membership-inference probes show retrain-comparable aggregate leakage, with a remaining gap under Min-K++.

1 INTRODUCTION

Machine unlearning has attracted growing interest as a means of removing the influence of specific training data from large language models — motivated by privacy regulations, data deletion requests, and the need to eliminate unsafe or unethical model behaviors (Bourtoule et al., 2021; Ginart et al., 2019). Retraining from scratch on the remaining data yields a model never exposed to the undesired data, but is prohibitively expensive for modern LLMs, which has driven the development of approximate unlearning methods that aim to efficiently achieve the same effect as if the model were retrained without the forget data.

Existing unlearning methods generally pursue two objectives: effectively erasing forget-set knowledge while preserving the model's performance on retained data. A substantial body of work (Jang et al., 2023; Yao et al., 2024; Zhang et al., 2024; Wang et al., 2025; Fan et al., 2025; Entesari et al., 2025) approaches this through fine-tuning, manipulating gradient signals to suppress memorized information while regularizing against retain-set degradation. More recently, several studies have shown that direct weight editing — originally developed for knowledge modification — can also serve as an effective and more efficient tool for knowledge removal (Fang et al., 2025; Shen et al., 2025).

Both families, however, share a deeper limitation in how they relate to the gold standard they aim to approximate. Loss-based methods such as GA, GD, and NPO optimize hand-designed objectives whose minima are not constrained to lie near the gold standard; because the loss only penalizes forget-set likelihood, the model is free to reach low loss through degenerate solutions that bear little resemblance to a model that has never seen the data. Editing and tuning-based methods such as AlphaEdit and LUNAR address this by specifying an explicit target for forget-set activations, but the target is a single vector, such as a fixed "unknown" embedding or an unlearning steering vector. A point target collapses the spread of forget-query activations onto one location, whereas a model that never saw the data answers different questions about an unfamiliar entity in varied ways. This leaves

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.

a question unanswered by both families: how can an unlearning objective capture the forget-query behavior of the gold-standard retrain model, rather than only suppressing likelihood or enforcing a point target?

To address this gap, we propose WADE (Wasserstein Distributional Alignment for Knowledge Erasure), a closed-form activation-editing method for LLM unlearning. WADE starts from the view that successful unlearning should match a retrain-like distribution over forget-query behavior. Since directly optimizing this output distribution is difficult, WADE realizes this goal through the internal MLP activation distribution that gives rise to the model's answers. Instead of minimizing a proxy forget loss or steering all forget examples to a single target vector, WADE aligns the distribution of forget activations with a target activation distribution associated with unknown-answer behavior. We construct this target distribution from unknown-entity prompts that preserve the original query format while removing the forgotten association. WADE then edits selected MLP down-projection matrices by minimizing a second-order Wasserstein discrepancy between the edited forget activations and the target activations, together with a retain-key regularizer. This discrepancy aligns both the mean and covariance of the activation distribution. Using the Bures–Procrustes form of covariance matching, WADE reduces the edit to a closed-form weight update, requiring no gradient-based training. This update has the normal-equation form of MEMIT (Meng et al., 2023); WADE's contribution lies in the distribution-derived constraints fed to it.

Our contributions are as follows:

- We introduce WADE, a closed-form activation-editing method that performs unlearning by aligning forget activations to a target activation distribution associated with unknown-answer behavior. This replaces proxy forget-loss optimization and single-vector activation steering with a distribution-level edit, allowing the method to model diverse forget-query behavior rather than forcing all forget examples toward the same point target.
- We derive a tractable closed-form edit for second-order distributional alignment. Using the Bures–Procrustes form of covariance matching, WADE converts mean-and-covariance alignment into a pseudo-sample least-squares problem with retain-key regularization, yielding a non-iterative weight update that provably does not increase the surrogate objective.
- We evaluate WADE on TOFU and WMDP-Cyber across five LLMs, with per-method hyperparameter sweeps, a K-FADE comparison, and capacity-matched controls. WADE gives the lowest KS-D to the retrain reference among evaluated baselines at published settings and the low-KS end of the tuned frontier at closed-form cost, but does not dominate the strongest baselines in utility.

2 Related Work

Optimization-based LLM unlearning. A common approach performs additional optimization on the original model. Gradient-ascent and gradient-difference methods reduce the likelihood of forget examples, often with retain-side regularization to limit utility degradation (Jang et al., 2023; Liu et al., 2022; Maini et al., 2024; Yao et al., 2024). Negative preference optimization improves the stability of direct likelihood suppression by casting unlearning as a preference-style objective (Zhang et al., 2024). Later variants simplify or regularize this objective, for example by removing reference-model dependence, deriving intrinsic token-probability preference signals, or imposing constrained entropic formulations (Fan et al., 2025; Qin et al., 2025; Entesari et al., 2025). Recent work also revisits second-order optimization for LLM unlearning. K-FADE uses K-FAC approximations to compute uphill Gauss–Newton updates, aiming to suppress forget-set outputs while minimizing changes on retain behavior and approximating retraining in output space (McKinney et al., 2026). Opt-Out regularizes gradient-based entity-level unlearning with a Wasserstein distance to the initial parameters (Choi et al., 2025); WADE instead aligns activation distributions and solves for the edit in closed form.

Representation-based and editing-based unlearning. A second family modifies internal representations or localized weights more directly. RMU reduces hazardous knowledge by steering intermediate representations away from their original behavior while preserving general capability (Li et al., 2024). LUNAR redirects unlearned-data activations toward regions associated with the model's

inability to answer, improving controllability under representation-level unlearning (Shen et al., 2025). A closely related line connects knowledge editing with unlearning. Recent work frames LLM unlearning as a special case of knowledge editing, where undesired information is modified into a refusal or empty-set response, and shows that editing methods can serve as strong unlearning baselines (Li et al., 2026). Under this view, methods such as ROME, MEMIT, WISE, and AlphaEdit can remove target knowledge through localized weight updates while preserving unrelated behavior (Meng et al., 2022; 2023; Wang et al., 2024; Fang et al., 2025; Li et al., 2026). However, most existing representation- and editing-based methods specify the desired forget behavior through a point target, steering direction, or target response, rather than matching a distribution of retrain-like behavior across forget queries.

Unlearning evaluation. TOFU evaluates whether an unlearned model behaves like a retrain-only reference on forget queries while preserving retain-set and general utility (Maini et al., 2024). OpenUnlearning further standardizes this protocol with metrics such as truth ratio, Forget Quality, KS distance, model utility, extraction strength, and membership-inference probes (Dorna et al., 2025). Among these metrics, FQ and KS distance directly compare the unlearned model's truth-ratio distribution with the retrain reference, making them suitable measures of retrain-reference alignment.

3 METHOD

3.1 PROBLEM FORMULATION

Let M denote a model obtained by fine-tuning a pretrained LLM on a dataset $\mathcal{D} = \mathcal{D}_r \cup \mathcal{D}_f$, where $\mathcal{D}_f$ is the forget set and $\mathcal{D}_r = \mathcal{D} \setminus \mathcal{D}_f$ is the retained data. Post-hoc machine unlearning starts from M and aims to produce an edited model M_{u} that removes the influence of $\mathcal{D}_f$ without retraining from scratch. The gold-standard reference is the retrain-only model M_{ret}, obtained by applying the same training recipe to $\mathcal{D}_r$ only, as if $\mathcal{D}_f$ had never been observed.

Classical unlearning definitions treat M_{ret} as the ideal deletion result: after unlearning, the resulting model should be indistinguishable from one trained without the deleted data (Cao & Yang, 2015; Ginart et al., 2019; Guo et al., 2020; Bourtoule et al., 2021). We write this ideal requirement as $M_{\mathrm{u}} \approx M_{\mathrm{ret}}$, where $\approx$ denotes behavioral indistinguishability under the relevant evaluation protocol.

3.2 CONSTRUCTING TARGET ACTIVATION DISTRIBUTION

Given the forget set $\mathcal{D}_f$, we construct a target prompt set $\mathcal{X}_t$ by identifying the entity or relation slots that define the knowledge to be forgotten, keeping the query template, and replacing only those slots with synthetic out-of-corpus values; many substitutions make $\mathcal{X}_t$ represent a distribution of unknown states. For factual QA, a question about a forgotten author becomes the same question about an unknown author name (Appendix A); for association-level forgetting of (s, r, o), only the subject of (s, r) queries is replaced, and queries about other relations of s join the retain set. No target answer, target activation, or per-example optimization is required.

We then forward $\mathcal{X}_t$ through the current model state and collect the MLP down-projection output activations at the edited token position. For QA data, the edit position is the final subword token of the metadata-matched entity span, with a final-prompt-token fallback. Let $h_l(x) \in \mathbb{R}^d$ denote the selected layer-l MLP output activation for prompt x. The target activation collection for layer l is

$$H_t^l = (h_l(x_i))_{i=1}^{n_t}, \qquad x_i \in \mathcal{X}_t. \tag{1}$$

It induces the empirical target distribution $P_t^l = \frac{1}{n_t} \sum_i \delta_{h_l(x_i)}$, where δ_z is the Dirac mass at z. P_t^l defines an unknown-state intervention: we do not assume that it reproduces the hidden states of M_{ret}, and retrain alignment is evaluated at the output level.

3.3 WASSERSTEIN DISTRIBUTIONAL ALIGNMENT

Prior editing studies provide evidence that MLP modules are effective intervention sites for factual associations: the MLP input can be viewed as a key that selects the relevant association, while the MLP output contributes the corresponding value to the residual stream (Meng et al., 2022;

2023). WADE therefore applies its distributional edit to the MLP down-projection matrix at selected transformer layers. The goal is to change how forget-set keys are mapped to MLP outputs, so that their output distribution matches the target activation distribution, while the mapping on retain-set keys is minimally perturbed.

For a layer l, let $W_l \in \mathbb{R}^{d\times m}$ denote the pre-update MLP down-projection matrix in the current model state. We collect its inputs at the selected token position for the forget and retain sets, denoted by $K_f^l = \{k_f^i\}_{i=1}^n$ and $K_r^l = \{k_r^j\}_{j=1}^{n_r}$, respectively. The corresponding target activation collection $H_t^l = (h_l(x_i))_{i=1}^{n_t}$ is constructed as in Eq. (1) and lies in the same output space as $W_l K_f^l$. We omit the layer superscript l when the layer is clear. Let $W' = W_l + \Delta W_l$, where ΔW_l denotes the weight update applied to the layer-l MLP down-projection. We formulate the layer-wise edit as

$$\min_{\Delta W_l} \mathcal{D}_2\big(P_{W'K_f}, P_t\big) + \lambda \operatorname{Tr}(\Delta W_l C_r \Delta W_l^\top) + \epsilon \|\Delta W_l\|_F^2, \tag{2}$$

where $P_{W'K_f}$ is the empirical distribution of edited forget activations $\{W'k_f^i\}_{i=1}^n$, P_t is the empirical distribution of target activations H_t, and C_r is the retain-key second moment,

$$C_r = \frac{1}{n_r}\sum_{j=1}^{n_r} k_r^j {k_r^j}^\top. \tag{3}$$

The first term aligns the edited forget activation distribution with the target distribution, while the second term limits changes to MLP outputs induced by retain keys, since $\operatorname{Tr}(\Delta W_l C_r \Delta W_l^\top) = \frac{1}{n_r}\sum_{j=1}^{n_r} \|\Delta W_l k_r^j\|_2^2$; the small ridge ϵ makes the solution unique (see below).

We instantiate $\mathcal{D}_2$ as a second-order Wasserstein moment surrogate that compares both the location and shape of two activation distributions. Given an empirical distribution P, let $\mu(P)$ and $\Sigma(P)$ denote its empirical mean and covariance, and let $\mathcal{G}(P) = \mathcal{N}(\mu(P), \Sigma(P))$ be the moment-matched Gaussian proxy. We define

$$\mathcal{D}_2(P, Q) = W_2^2(\mathcal{G}(P), \mathcal{G}(Q)) = \|\mu(P) - \mu(Q)\|_2^2 + \mathcal{B}^2(\Sigma(P), \Sigma(Q)), \tag{4}$$

where $\mathcal{B}$ is the Bures metric between covariance matrices. This surrogate only uses empirical first and second moments and does not assume Gaussian activations; by Gelbrich's bound it lower-bounds $W_2^2(P, Q)$ (Gelbrich, 1990), so it controls only the first two moments. Let (μ_f, Σ_f) be the empirical mean and covariance of the forget keys K_f, and let (μ_t, Σ_t) be those of the target activations H_t. Since the edited down-projection is linear, the edited forget activations have mean $W'\mu_f$ and covariance $W'\Sigma_f {W'}^\top$. Eq. (2) can thus be written as

$$\min_{\Delta W_l} \|W'\mu_f - \mu_t\|_2^2 + \mathcal{B}^2(W'\Sigma_f {W'}^\top, \Sigma_t) + \lambda \operatorname{Tr}(\Delta W_l C_r \Delta W_l^\top) + \epsilon \|\Delta W_l\|_F^2. \tag{5}$$

To make the update tractable and closed-form, we use the Procrustes representation of the Bures covariance term. With covariance factors, the Bures distance can be written as a minimum over orthogonal alignments between factors. In practice, we first compute adaptive low-rank decompositions for the forget-key and target-activation covariances. If their ranks differ, we set $r = \min(r_f, r_t)$ and retain the first r principal directions of both factors. With the resulting common-rank factors $\Sigma_f \approx U\Lambda U^\top$ and $\Sigma_t \approx V\Gamma V^\top$, we obtain

$$\mathcal{B}^2(W'\Sigma_f {W'}^\top, \Sigma_t) \approx \min_{O\in\mathcal{O}(r)} \|W'U\Lambda^{1/2} - V\Gamma^{1/2}O\|_F^2. \tag{6}$$

The approximation comes from covariance truncation, while the orthogonal matrix O accounts for the rotational non-uniqueness of covariance factors.

Combining the mean and covariance factors gives the pseudo-sample matrices

$$\widetilde{K} = [\mu_f,\ U\Lambda^{1/2}], \qquad \widetilde{H}(O) = [\mu_t,\ V\Gamma^{1/2}O]. \tag{7}$$

The distributional term is then approximated by a least-squares alignment between these pseudo samples:

$$\mathcal{D}_2\big(P_{W'K_f}, P_t\big) \approx \min_{O\in\mathcal{O}(r)} \|W'\widetilde{K} - \widetilde{H}(O)\|_F^2. \tag{8}$$

The variables ΔW_l and O are coupled in this objective. WADE uses a one-step Procrustes decoupling: it fixes the covariance-factor orientation under the pre-edit map W_l, and then solves the resulting retain-regularized least-squares problem in closed form. Specifically, we choose

$$O^\star = \arg\min_{O \in \mathcal{O}(r)} \|W_l U \Lambda^{1/2} - V \Gamma^{1/2} O\|_F^2. \tag{9}$$

If $M_O = \Gamma^{1/2} V^\top W_l U \Lambda^{1/2} = X \Sigma_O Y^\top$, the Procrustes solution is $O^\star = XY^\top$. With this orientation fixed, let $\widetilde{H}^\star = \widetilde{H}(O^\star)$. The reduced objective becomes

$$\min_{\Delta W_l} \|\Delta W_l \widetilde{K} - (\widetilde{H}^\star - W_l \widetilde{K})\|_F^2 + \lambda \operatorname{Tr}(\Delta W_l C_r \Delta W_l^\top) + \epsilon \|\Delta W_l\|_F^2, \tag{10}$$

which yields

$$\Delta W_l = \big(\widetilde{H}^\star - W_l \widetilde{K}\big) \widetilde{K}^\top \big(\widetilde{K} \widetilde{K}^\top + \lambda C_r + \epsilon I\big)^{-1}. \tag{11}$$

The ridge is structural: $\widetilde{K}\widetilde{K}^\top + \lambda C_r$ has rank at most $r+1+n_r$ and is singular whenever this is below the MLP width m. Eq. (11) has the normal-equation form of MEMIT's batch editor (Meng et al., 2023) with keys $\widetilde{K}$, residual $\widetilde{H}^\star - W_l \widetilde{K}$, and key covariance $\lambda C_r + \epsilon I$; WADE differs in deriving $r+1$ pseudo key–value pairs from distribution moments instead of one optimized value per fact, and the mean-only variant is the rank-one case $r = 0$. Since the fixed-$O^\star$ objective upper-bounds the surrogate and is tight at $\Delta W_l = 0$, the update never increases the surrogate objective (Proposition 1; derivation and proof in Appendix C).

3.4 Sequential Multi-Layer Editing

WADE applies the layer-wise update in Eq. (11) sequentially to an ordered set of edited layers $\mathcal{L} = (l_1, \ldots, l_s)$. We discuss layer selection in Sec. 4.3. Let $M^{(0)} = M$. At step i, WADE forwards the forget, retain, and target prompts through the current model $M^{(i-1)}$, and collects $K_f^{l_i}$, $K_r^{l_i}$, and $H_t^{l_i}$ at layer l_i. The prompt sets are fixed, but their activations are re-collected under the current model state at each edited layer. It then computes ΔW_{l_i} using Eq. (11) and updates only the corresponding down-projection matrix,

$$W_{l_i}^{(i)} = W_{l_i}^{(i-1)} + \Delta W_{l_i}. \tag{12}$$

This is not a joint optimization over layers: each layer solves its own closed-form problem under the representation distribution induced by previous edits, so later updates adapt to earlier ones, whereas a joint solve would lose the layer-wise closed form. The shallow-to-deep order is part of the solver (Sec. 4.3).

4 Experiments

We evaluate whether WADE matches the retrain reference while preserving utility (Sec. 4.2), whether full distributional alignment is necessary (Sec. 4.3), and whether its alignment holds under membership inference and on WMDP-Cyber (Sec. 4.4).

4.1 Experimental Setup

Datasets We evaluate WADE on two complementary settings. TOFU (Maini et al., 2024; Dorna et al., 2025) is an instance-level factual unlearning benchmark built from synthetic author biographies, and serves as our primary testbed for baseline comparison and retrain-reference matching. WMDP-Cyber (Li et al., 2024) measures hazardous cyber knowledge in multiple-choice form, and serves as a transfer check for whether the same activation-editing principle reduces undesirable domain knowledge.

Reference prompt datasets. WADE collects target activations from task-specific reference prompts. For TOFU, we use template-matched unknown-author prompts obtained by replacing each forgotten author with a synthetic out-of-corpus name; an example is provided in Appendix A. For WMDP-Cyber, we use harmful instruction prompts from prior refusal-direction work (Arditi et al., 2024) to induce refusal-aligned target activations.

Models The main TOFU comparison uses Llama-3.2-3B, Llama-2-7B-Chat, and Qwen3-8B, covering different model families, scales, and memorization behaviors. The ablations further include Mistral-Nemo-12B and Phi-4-14B, and WMDP-Cyber uses Zephyr-7B-beta to match prior hazardous-knowledge experiments.

Metrics On TOFU, we follow the Open-Unlearning evaluator (Dorna et al., 2025) and report truth ratio (TR), Forget Quality (FQ), KS distance (KS-D), and model utility (MU). FQ and KS-D are computed by a two-sample Kolmogorov–Smirnov test comparing the unlearned model's forget-set TR distribution with the retrain reference; KS-D is the test statistic (lower is better), and FQ is the corresponding p-value (higher is better). At fixed sample sizes FQ is a monotone function of KS-D, so the two form one criterion. With 400 truth ratios per side, KS-D is a multiple of $1/400$; we report it rounded half up to three decimals, and FQ is computed from the unrounded statistic.

Baselines We compare WADE against reference checkpoints and recent unlearning methods. Finetuned denotes the original model before unlearning, and Retrain denotes the split-matched retrain-only reference. Algorithmic baselines include GA (Jang et al., 2023), GD (Liu et al., 2022; Maini et al., 2024), NPO (Zhang et al., 2024), SimNPO (Fan et al., 2025), RMU (Li et al., 2024), LUNAR (Shen et al., 2025), and AlphaEdit (Fang et al., 2025), using their released per-model configurations. Appendix F adds per-method hyperparameter sweeps and K-FADE (McKinney et al., 2026).

Implementation details Unless otherwise specified, WADE edits MLP down-projection matrices in layers 6–8 with $\lambda = 0.01$ and $\epsilon = 10^{-4}$. The window was selected once on Llama-2-7B-Chat forget10 (Sec. 4.3) and is reused without per-model search for all other models, splits, and WMDP-Cyber. Covariance factors use adaptive PCA ranks retaining 99% explained variance; rank mismatch in the Procrustes step is handled by $r = \min(r_f, r_t)$, with no rank cap by default. TOFU activations use the final subword token of the matched author span, the first causal position that sees the complete name, with final-prompt-token fallback.

4.2 Main Results

Table 1: TOFU forget10 main results. Bold marks the best unlearning result on retrain-reference alignment metrics; the retrain checkpoint is shown only as the reference.

	Llama-3.2-3B				**Llama-2-7B**				**Qwen3-8B**			
Method	TR	FQ↑	KS-D↓	MU↑	TR	FQ↑	KS-D↓	MU↑	TR	FQ↑	KS-D↓	MU↑
Finetuned	0.473	3.60e-27	0.388	0.665	0.518	4.35e-25	0.373	0.628	0.438	6.16e-18	0.315	0.637
Retrain	0.644	1.000	0.000	0.650	0.681	1.000	0.000	0.613	0.563	1.000	0.000	0.683
GA	0.000	1.06e-239	1.000	0.000	0.000	1.06e-239	1.000	0.000	0.000	1.06e-239	1.000	0.000
GD	0.000	8.51e-237	0.998	0.538	0.000	8.51e-237	0.998	0.646	0.369	6.16e-18	0.315	0.599
SimNPO	0.495	2.21e-19	0.328	0.658	0.555	1.49e-16	0.303	0.619	0.472	1.03e-14	0.285	0.610
NPO	0.610	0.030	0.103	0.630	0.610	0.037	0.100	0.621	0.449	1.49e-16	0.303	0.591
RMU	0.635	3.65e-11	0.248	0.646	0.604	1.10e-9	0.230	0.602	0.552	5.97e-8	0.208	0.613
AlphaEdit	0.672	1.82e-26	0.383	0.619	0.598	1.79e-13	0.273	0.627	0.401	3.20e-18	0.318	0.645
LUNAR	0.583	3.09e-6	0.183	0.459	0.655	0.013	0.113	0.564	0.559	9.91e-11	0.243	0.562
WADE (ours)	0.651	**0.094**	**0.088**	0.624	0.661	**0.758**	**0.048**	0.578	0.584	**1.76e-3**	**0.133**	0.592

Table 1 shows that, at published baseline settings, WADE attains the lowest KS-D, and hence the highest FQ, among unlearning methods on all three models. On Llama-2-7B, WADE reaches KS-D $= 0.048$ (FQ $= 0.758$), against 0.100 (FQ $= 0.037$) for NPO and 0.113 (FQ $= 0.013$) for LUNAR; on Llama-3.2-3B and Qwen3-8B it reaches 0.088 and 0.133. At the 0.05 level, WADE is the only unlearning method whose truth-ratio distribution is not rejected as different from retrain on Llama-3.2-3B and Llama-2-7B; on Qwen3-8B all methods are rejected. Several baselines have mean TR close to retrain yet large KS-D (e.g., AlphaEdit on Llama-3.2-3B), so matching average behavior does not imply matching its distribution. WADE does not lead in utility: its MU is 0.04–0.05 below the finetuned model on all three models. Under per-method sweeps on Llama-2-7B (Appendix F), the pooled frontier consists of WADE at the low-KS end (KS-D 0.038, MU 0.580) and NPO at the high-utility end (0.088, 0.625), and all RMU and LUNAR configurations are dominated. On Llama-3.2-3B, K-FADE's best step sizes bracket WADE without dominating it (e.g., KS-D 0.073 at

MU 0.619) but need about 664 s of curvature fitting, versus about 16.5 s for WADE's three-layer edit (Appendix B); relative to K-FADE, WADE's advantage is efficiency at comparable alignment.

4.3 Ablation Studies

We follow the baseline TOFU setup and ablate the target statistic used by WADE. The mean-only variant uses a single target vector, the target activation mean μ_t, and ignores covariance. The distributional variant matches both the mean μ_t and covariance Σ_t. For each model, the two variants use the same reference prompts, edited layers, and retain regularization strength; only the target statistic differs. This ablation tests whether WADE's gains come from full distributional alignment rather than steering forget activations toward a point target.

Table 2: Effect of target statistics. Matching both mean and covariance consistently improves retrain-reference alignment over mean-only alignment across five models.

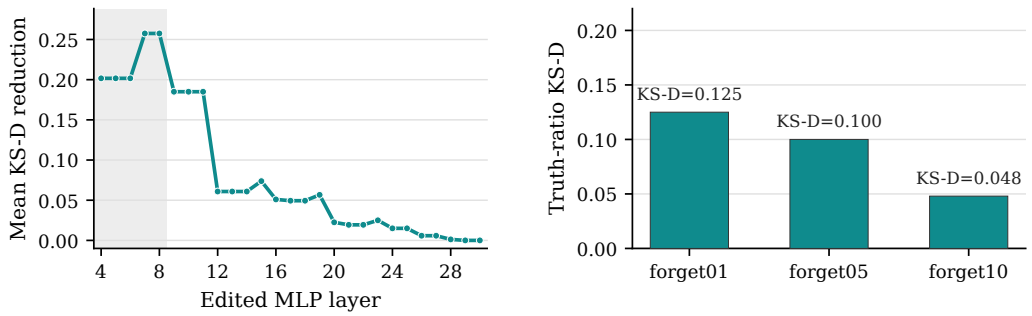
	Base			After unlearning			
Model	TR	MU	Target	TR	FQ↑	KS-D↓	MU↑
Llama-2-7B-Chat	0.518	0.628	Mean-only	0.725	1.10e-9	0.230	0.223
			Distributional	0.661	0.758	0.048	0.578
Llama-3.2-3B	0.473	0.665	Mean-only	0.553	2.63e-10	0.238	0.540
			Distributional	0.651	0.094	0.088	0.624
Qwen3-8B	0.438	0.637	Mean-only	0.549	1.07e-8	0.218	0.471
			Distributional	0.584	1.76e-3	0.133	0.592
Mistral-Nemo-12B	0.458	0.728	Mean-only	0.507	2.77e-9	0.225	0.633
			Distributional	0.519	5.83e-4	0.143	0.654
Phi-4-14B	0.436	0.386	Mean-only	0.524	1.65e-8	0.215	0.562
			Distributional	0.542	6.92e-5	0.160	0.573

Table 2 shows that matching only the target mean is insufficient for retrain-reference alignment. Across all five models, the mean-only variant yields very low FQ values and KS-D between 0.215 and 0.238, indicating a clear mismatch from the retrain reference. Adding covariance alignment consistently improves both distributional metrics: FQ increases by several orders of magnitude, and KS-D drops to 0.048–0.160. These results support the core design choice of WADE: aligning the target activation distribution is more effective than moving forget activations toward a mean target alone.

Since mean-only alignment is the rank-one case of WADE, Appendix F adds capacity controls on Llama-2-7B. Scaling the mean-only update to WADE's total norm worsens alignment (KS-D 0.345, MU collapses), covariance-only alignment aligns weakly (0.268), and a same-spectrum random rotation of the target covariance reaches KS-D 0.082 but lowers MU to 0.442. Update norm, rank, and spectrum alone therefore do not explain the gain. A paired-regression variant that maps each forget key to its own constructed target is competitive (KS-D 0.063, MU 0.586) and neither dominates, but it relies on a construction-induced pairing that WADE does not need. Across five disjoint sets of fictitious names, KS-D varies within 0.010.

We next examine two practical factors on Llama-2-7B-Chat: edit location and forget-set size. For edit location, we sweep fixed-width contiguous MLP edit windows under the same WADE setting and score each window by post-edit KS-D to the retrain reference. Each layer is assigned the average KS-D reduction over the swept windows that contain it. For forget-set size, we evaluate WADE on TOFU forget01, forget05, and forget10 against the corresponding split-matched retrain reference.

Figure 1(a) shows that WADE's strongest edit region is concentrated in shallow MLP layers. Layers 4–8 yield the largest average KS-D reductions, with mean reductions above 0.20, while later layers give smaller gains. We therefore use layers 6–8 as a compact default edit window in the main experiments, which remains within this high-gain shallow-layer region while reducing the number of edited layers. This profile is specific to the ascending solver: descending and random edit orders give KS-D 0.148 and 0.113 (3-seed mean). Because the window was selected on the reported forget10 split, the other models and the forget01/05 splits serve as held-out checks. Figure 1(b) shows that WADE remains



(a) Edit-location analysis.

(b) TOFU forget-split analysis.

Figure 1: Layer and forget-size analyses on Llama-2-7B-Chat. WADE is most effective in shallow MLP layers and remains close to split-matched retrain references across TOFU forget sizes.

close to the split-matched retrain reference across TOFU forget sizes, with KS-D values of 0.125, 0.100, and 0.048 on forget01, forget05, and forget10, respectively. The corresponding FQ values are 0.919, 0.270, and 0.758, so the FQ test does not reject retrain-reference alignment on any of the three splits. Additional sensitivity analyses in Appendix E show that WADE is stable over a moderate range of retain regularization strengths and that the uncapped adaptive covariance rank can be closely approximated by a moderate rank cap.

4.4 Safety-Oriented Evaluation

Membership inference. We use membership inference as the main safety-oriented probe to test whether forget examples remain distinguishable after unlearning. We evaluate the Llama-2-7B-Chat TOFU forget10 setting, using forget10 examples as members and holdout10 examples as nonmembers. The attack suite includes sequence loss, Min-K (Shi et al., 2024), Min-K++ (Zhang et al., 2025), and zlib-normalized loss (Carlini et al., 2021). For each attack, we compute AUC under the member and nonmember labels, and report MIA leak as the mean $|\mathrm{AUC} - 0.5|$ across attacks. We also compare each method's forget-example attack-score distribution with the retain90 retrain reference using KS distance and z-normalized Wasserstein-1 distance, averaged over the attack suite. TR-KS is reported to connect the attack-based probe with the truth-ratio alignment used in the main evaluation.

Table 3: Membership-inference evaluation on TOFU forget10. WADE achieves retrain-comparable aggregate leakage and the lowest average attack-score distance to the retrain reference in this comparison, with per-attack differences reported in Appendix D.

Method	MIA leak↓	MIA-score KS↓	MIA-score W1z↓	TR-KS↓	MU↑
Retrain (reference)	0.101	0.000	0.000	0.000	0.613
Finetuned	0.499	0.994	2.044	0.375	0.628
NPO	0.394	0.543	18.582	0.100	0.621
LUNAR	0.171	0.340	0.654	0.115	0.564
WADE (ours)	0.099	0.308	0.617	0.048	0.578

Table 3 shows that WADE is closest to the retrain reference under the aggregate MIA summary. Its aggregate MIA leak is 0.099, close to the retrain reference at 0.101, while Finetuned, NPO, and LUNAR reach 0.499, 0.394, and 0.171, respectively. WADE also gives the smallest average attack-score distance to the retrain reference among unlearning methods, with MIA-score KS $= 0.308$ and W1z $= 0.617$. Its TR-KS is also the lowest at 0.048. The aggregate hides one failure mode: under Min-K++, retrain is at chance ($|\mathrm{AUC} - 0.5| = 0.002$) while WADE remains separable (0.287), though less so than NPO and LUNAR. We therefore claim retrain-comparable leakage in aggregate, not per attack (Appendix D).

WMDP-Cyber. On Zephyr-7B-beta, forget keys are collected from WMDP-Cyber questions, retain keys are collected from up to 500 raw text chunks sampled from the WMDP cyber retain corpus at the final valid token position, and target activations are collected from harmful instruction prompts used in prior refusal-direction work (Arditi et al., 2024) to induce refusal-aligned states. The evaluator is zero-shot multiple choice; lower accuracy indicates less hazardous-knowledge retention. We also report MMLU and its `college_computer_science` (MMLU-CS) and `computer_security` (MMLU-Sec) subjects as utility checks.

Table 4: WMDP-Cyber on Zephyr-7B-beta (chance accuracy 0.25 for all columns). Among methods that keep MMLU at the base level, WADE gives the largest WMDP-Cyber reduction. †LUNAR's MMLU falls to chance, so its reduction is not selective.

Method	WMDP-Cyber↓	Δ vs. base↑	MMLU↑	MMLU-CS↑	MMLU-Sec↑
Base (unedited)	0.410	–	0.460	0.350	0.590
NPO	0.400	0.010	0.459	0.370	0.590
RMU	0.315	0.095	0.458	0.370	0.490
LUNAR†	0.264	0.146	0.248	0.250	0.220
WADE (ours)	0.277	**0.133**	**0.462**	**0.370**	0.520

Table 4 shows that WADE reduces WMDP-Cyber accuracy from 0.410 to 0.277 while its MMLU stays at the base level (0.462 vs. 0.460). Among methods that preserve general utility, this is the largest reduction: 0.133 for WADE versus 0.095 for RMU and 0.010 for NPO. LUNAR reduces WMDP-Cyber slightly more, but its MMLU falls to 0.248, i.e., chance level, so its reduction reflects general degradation rather than selective forgetting. The security subject drops from 0.590 to 0.520 under WADE, less than under RMU (0.490); prior work observes similar drops on closely related MMLU subjects for RMU (Li et al., 2024).

5 Limitations

WADE has four main limitations. First, it defines post-unlearning behavior through a target prompt set; we evaluate entity substitution (TOFU) and refusal prompts (WMDP-Cyber), and other settings, including association-level forgetting, need their own slot specification. The target defines an unknown-state intervention and is not validated as a proxy for retrain hidden states. Second, the two-moment surrogate lower-bounds W_2 and does not control higher-order or tail behavior, which may relate to the Min-K++ gap; WADE provides no certified deletion. Third, alternatives to the single Procrustes step, common-rank truncation, and per-layer target re-collection were not evaluated. Fourth, the edit window was selected on the reported Llama-2-7B split, sweeps and controls use one model each, and WADE does not dominate the strongest baselines in utility; alignment is also weaker on the 8B–14B models.

6 Conclusion

We propose WADE, a closed-form activation-editing method for LLM unlearning that replaces point-target steering with distributional alignment: it matches the mean and covariance of forget-query MLP activations to an unknown-answer target distribution and reduces this objective, through Bures–Procrustes pseudo-samples, to a MEMIT-form weight update that provably does not increase the surrogate. On TOFU, WADE gives the lowest KS distance to the retrain reference among evaluated baselines at published settings and the low-KS end of the tuned frontier at a fraction of the cost of iterative or curvature-based methods, while NPO retains an advantage in utility. Capacity-matched controls attribute the gain to target-specific covariance structure, membership-inference probes support retrain-comparable leakage in aggregate but expose a remaining Min-K++ gap, and WMDP-Cyber results show transfer beyond TOFU.

AI USE STATEMENT

We used generative AI tools to assist with manuscript revision and language polishing. The authors reviewed all AI-assisted text and take full responsibility for the final manuscript.

ETHICS STATEMENT

Unlearning is motivated by privacy and safety. WADE is an approximate method without formal deletion guarantees, and our membership-inference results show that individual attacks can still separate forget data; it should not be used as a substitute for retraining where legal guarantees are required. WMDP-Cyber experiments use the public benchmark and prompts from prior work, and we release no new hazardous content.

REPRODUCIBILITY STATEMENT

Sec. 3 and Appendix C give the full update, Sec. 4.1 and Appendix F give all WADE hyperparameters, baseline grids, and the K-FADE setup. Anonymized code for WADE is provided in the supplementary material.

REFERENCES

Andy Arditi, Oscar Obeso, Aaquib Syed, Daniel Paleka, Nina Rimsky, Wes Gurnee, and Neel Nanda. Refusal in language models is mediated by a single direction. *arXiv preprint arXiv:2406.11717*, 2024. doi: 10.48550/arXiv.2406.11717.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A. Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159. IEEE, 2021. doi: 10.1109/SP40001.2021.00019.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *2015 IEEE Symposium on Security and Privacy*, pp. 463–480. IEEE Computer Society, 2015. doi: 10.1109/SP.2015.35.

Nicholas Carlini, Florian Tramèr, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Úlfar Erlingsson, Alina Oprea, and Colin Raffel. Extracting training data from large language models. In *30th USENIX Security Symposium*, 2021.

Minseok Choi, Daniel Rim, Dohyun Lee, and Jaegul Choo. Opt-out: Investigating entity-level unlearning for large language models via optimal transport. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (ACL)*, 2025.

Vineeth Dorna, Anmol Mekala, Wenlong Zhao, Andrew McCallum, Zachary C Lipton, J Zico Kolter, and Pratyush Maini. OpenUnlearning: Accelerating LLM unlearning via unified benchmarking of methods and metrics. *arXiv preprint arXiv:2506.12618*, 2025. URL https://arxiv.org/abs/2506.12618.

Taha Entesari, Arman Hatami, Rinat Khaziev, Anil Ramakrishna, and Mahyar Fazlyab. Constrained entropic unlearning: A primal-dual framework for large language models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2506.05314.

Chongyu Fan, Jiancheng Liu, Licong Lin, Jinghan Jia, Ruiqi Zhang, Song Mei, and Sijia Liu. Simplicity prevails: Rethinking negative preference optimization for LLM unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2410.07163.

Junfeng Fang, Houcheng Jiang, Kun Wang, Yunshan Ma, Jie Shi, Xiang Wang, Xiangnan He, and Tat-Seng Chua. AlphaEdit: Null-space constrained knowledge editing for language models. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.02355.

Matthias Gelbrich. On a formula for the L^2 Wasserstein metric between measures on Euclidean and Hilbert spaces. *Mathematische Nachrichten*, 147(1):185–203, 1990.

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y. Zou. Making AI forget you: Data deletion in machine learning. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3832–3842. PMLR, 2020.

Joel Jang, Dongkeun Yoon, Sohee Yang, Sungmin Cha, Moontae Lee, Lajanugen Logeswaran, and Minjoon Seo. Knowledge unlearning for mitigating privacy risks in language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (ACL)*, pp. 14389–14408. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.805. arXiv:2210.01504.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Gabriel Mukobi, et al. The WMDP benchmark: Measuring and reducing malicious use with unlearning. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235 of *Proceedings of Machine Learning Research*, pp. 28525–28550. PMLR, 2024. arXiv:2403.03218.

Zexi Li, Xiangzhu Wang, William F. Shen, Meghdad Kurmanji, Xinchi Qiu, Dongqi Cai, Chao Wu, and Nicholas D. Lane. Editing as unlearning: Are knowledge editing methods strong baselines for large language model unlearning? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 37627–37635, 2026. doi: 10.1609/aaai.v40i44.41097.

Bo Liu, Qiang Liu, and Peter Stone. Continual learning and private unlearning. In *Conference on Lifelong Learning Agents (CoLLAs)*, pp. 243–254. PMLR, 2022.

Pratyush Maini, Zhili Feng, Avi Schwarzschild, Zachary Chase Lipton, and J. Zico Kolter. TOFU: A task of fictitious unlearning for LLMs. In *First Conference on Language Modeling (COLM)*, 2024. arXiv:2401.06121.

Lev McKinney, Anvith Thudi, Juhan Bae, Tara Rezaei Kheirkhah, Nicolas Papernot, Sheila A. McIlraith, and Roger B. Grosse. Gauss-Newton unlearning for the LLM era. *arXiv preprint arXiv:2602.10568*, 2026.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. arXiv:2202.05262.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.07229.

Kai Qin, Jiaqi Wu, Jianxiang He, Haoyuan Sun, Yifei Zhao, Bin Liang, Yongzhe Chang, Tiantian Zhang, and Houde Liu. Distribution preference optimization: A fine-grained perspective for LLM unlearning. *arXiv preprint arXiv:2510.04773*, 2025.

William F. Shen, Xinchi Qiu, Meghdad Kurmanji, Alex Iacob, Lorenzo Sani, Yihong Chen, Nicola Cancedda, and Nicholas D. Lane. LUNAR: LLM unlearning via neural activation redirection. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. arXiv:2502.07218.

Weijia Shi, Anirudh Ajith, Mengzhou Xia, Yangsibo Huang, Daogao Liu, Terra Blevins, Danqi Chen, and Luke Zettlemoyer. Detecting pretraining data from large language models. In *International Conference on Learning Representations (ICLR)*, 2024.

Peng Wang, Zexi Li, Ningyu Zhang, Ziwen Xu, Yunzhi Yao, Yong Jiang, Pengjun Xie, Fei Huang, and Huajun Chen. WISE: Rethinking the knowledge memory for lifelong model editing of large language models. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Qizhou Wang, Jin Peng Zhou, Zhanke Zhou, Saebyeol Shin, Bo Han, and Kilian Q. Weinberger. Rethinking LLM unlearning objectives: A gradient perspective and go beyond. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025. arXiv:2502.19301.

Yuanshun Yao, Xiaojun Xu, and Yang Liu. Large language model unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2310.10683.

Jingyang Zhang, Jingwei Sun, Eric Yeats, Yang Ouyang, Martin Kuo, Jianyi Zhang, Hao Frank Yang, and Hai Li. Min-K%++: Improved baseline for detecting pre-training data from large language models. In *International Conference on Learning Representations (ICLR)*, 2025.

Ruiqi Zhang, Licong Lin, Yu Bai, and Song Mei. Negative preference optimization: From catastrophic collapse to effective unlearning. In *First Conference on Language Modeling (COLM)*, 2024. arXiv:2404.05868.

APPENDIX

A TARGET PROMPT CONSTRUCTION

TOFU example. For factual QA unlearning, WADE constructs target prompts by preserving the question template and replacing only the forgotten entity with a synthetic out-of-corpus entity. These prompts are used to induce unknown-entity activation states during target activation collection; the response text below is shown only to illustrate the intended state and is not used as a supervised target.

TOFU factual QA target construction

Forget prompt: Who is `[forgotten author]` and what are they known for?

Original-model response: *`[forgotten author]` is a contemporary novelist known for their award-winning fiction and distinctive narrative style.*

Target prompt: Who is Aleksandra Petrov-Hwanwoo and what are they known for?

Target-state response: *I do not have reliable information about Aleksandra Petrov-Hwanwoo. This appears to be an unknown or fictional author name.*

Association-level construction. To forget only the association (s, r, o) (e.g., an author's birthplace) while keeping other facts about s, forget prompts are queries $x(s, r)$, target prompts are $x(\tilde{s}, r)$ for synthetic entities $\tilde{s}$, and retain prompts include queries $x(s, r')$ for $r' \neq r$. The objective and solver are unchanged. We have not evaluated this variant.

Sensitivity to target names. We built five target sets from non-overlapping pools of fictitious names, holding templates, number of prompts, token positions, and layers fixed. On Llama-2-7B-Chat forget10, KS-D varies within 0.010 and MU within 0.0062 across the five sets, and the edited models' responses remain consistent with unknown-answer behavior.

B EDITING OVERHEAD

Table 5 reports wall-clock editing overhead on TOFU. For AlphaEdit, LUNAR, and WADE, we report measured time per edited layer. For RMU, NPO, GA, and GD, which do not have a per-layer editing notion, we report estimated total time for a 5-epoch run. These estimates use the measured Llama-2-7B runtime as the anchor and scale linearly with model size. Downstream evaluation time is excluded. WADE's default three-layer edit takes about 16.5 s on Llama-3.2-3B and 21.3 s on Llama-2-7B. For K-FADE on Llama-3.2-3B, fitting K-FAC factors over the 84 gate/up/down projections took about 664 s for factor fitting alone, excluding the update and evaluation.

Table 5: Editing overhead in seconds on TOFU across model scales.

Method	Llama-3.2-3B	Llama-2-7B	Mistral-Nemo-12B	Phi-4-14B
Time per edited layer				
AlphaEdit	5.9	7.2	10.1	12.0
LUNAR	35.2	31.7	39.5	43.5
WADE (ours)	**5.5**	**7.1**	**8.1**	**10.7**
Time for 5 epochs				
RMU	62.4	145.6	249.6	291.2
NPO	213.3	497.8	853.4	995.6
GA	83.6	195.1	334.5	390.2
GD	179.9	419.8	719.7	839.6

C CLOSED-FORM SOLUTION OF WADE

We derive the closed-form layer-wise update used in WADE. For a fixed edited layer, we omit the layer index l. Let $W \in \mathbb{R}^{d\times m}$ be the pre-update MLP down-projection at the current editing step, and let $W' = W + \Delta W$. WADE solves

$$\min_{\Delta W} \|W'\mu_f - \mu_t\|_2^2 + \mathcal{B}^2(W'\Sigma_f W'^\top, \Sigma_t) + \lambda \operatorname{Tr}(\Delta W C_r \Delta W^\top) + \epsilon\|\Delta W\|_F^2, \tag{13}$$

where μ_f, Σ_f are the empirical mean and covariance of forget keys, μ_t, Σ_t are the empirical mean and covariance of target activations, and C_r is the retain-key second moment.

Here r denotes the common Procrustes rank defined in Sec. 3.3. Let the low-rank covariance factors be

$$\Sigma_f \approx U\Lambda U^\top, \qquad \Sigma_t \approx V\Gamma V^\top, \tag{14}$$

with $U \in \mathbb{R}^{m\times r}$, $V \in \mathbb{R}^{d\times r}$, and diagonal $\Lambda, \Gamma \succeq 0$. Then

$$W'\Sigma_f W'^\top \approx (W'U\Lambda^{1/2})(W'U\Lambda^{1/2})^\top, \qquad \Sigma_t \approx (V\Gamma^{1/2})(V\Gamma^{1/2})^\top. \tag{15}$$

Using the Bures–Procrustes factorization with these low-rank factors,

$$\mathcal{B}^2(W'\Sigma_f W'^\top, \Sigma_t) \approx \min_{O\in\mathcal{O}(r)} \left\| W'U\Lambda^{1/2} - V\Gamma^{1/2}O \right\|_F^2, \tag{16}$$

where $\mathcal{O}(r) = \{O \in \mathbb{R}^{r\times r} : O^\top O = I\}$. The approximation comes from the low-rank covariance truncation; with full covariance factors, the Procrustes representation is exact. Thus the distributional alignment term can be written as

$$\|W'\mu_f - \mu_t\|_2^2 + \left\| W'U\Lambda^{1/2} - V\Gamma^{1/2}O \right\|_F^2 = \left\| W'\widetilde{K} - \widetilde{H}(O) \right\|_F^2, \tag{17}$$

where

$$\widetilde{K} = \begin{bmatrix} \mu_f & U\Lambda^{1/2} \end{bmatrix} \in \mathbb{R}^{m\times(r+1)}, \qquad \widetilde{H}(O) = \begin{bmatrix} \mu_t & V\Gamma^{1/2}O \end{bmatrix} \in \mathbb{R}^{d\times(r+1)}. \tag{18}$$

WADE fixes O using the pre-update weight W at the current editing step. Let

$$A = WU\Lambda^{1/2}, \qquad B = V\Gamma^{1/2}. \tag{19}$$

The orthogonal factor is chosen by

$$O^\star = \arg\min_{O\in\mathcal{O}(r)} \|A - BO\|_F^2. \tag{20}$$

Expanding the objective gives

$$\|A - BO\|_F^2 = \operatorname{Tr}(A^\top A) + \operatorname{Tr}(O^\top B^\top BO) - 2\operatorname{Tr}(O^\top B^\top A) \tag{21}$$

$$= \operatorname{Tr}(A^\top A) + \operatorname{Tr}(B^\top B) - 2\operatorname{Tr}(O^\top B^\top A), \tag{22}$$

where the second equality uses the orthogonality of O. Therefore Eq. (20) is equivalent to

$$O^\star = \arg\max_{O\in\mathcal{O}(r)} \operatorname{Tr}(O^\top B^\top A). \tag{23}$$

Let

$$B^\top A = \Gamma^{1/2}V^\top WU\Lambda^{1/2} = X\Sigma_O Y^\top \tag{24}$$

be its singular value decomposition. Then the orthogonal Procrustes solution is

$$O^\star = XY^\top. \tag{25}$$

With $O^\star$ fixed, define

$$\widetilde{H}^\star = \widetilde{H}(O^\star). \tag{26}$$

The layer-wise WADE objective becomes

$$\min_{\Delta W} \left\| (W + \Delta W)\widetilde{K} - \widetilde{H}^\star \right\|_F^2 + \lambda \operatorname{Tr}(\Delta W C_r \Delta W^\top) + \epsilon\|\Delta W\|_F^2. \tag{27}$$

Let

$$E = \widetilde{H}^\star - W\widetilde{K}. \tag{28}$$

Then Eq. (27) is

$$\min_{\Delta W} \left\|\Delta W\widetilde{K} - E\right\|_F^2 + \lambda \operatorname{Tr}(\Delta W C_r \Delta W^\top) + \epsilon\|\Delta W\|_F^2. \tag{29}$$

Expanding the first term,

$$\left\|\Delta W\widetilde{K} - E\right\|_F^2 = \operatorname{Tr}\left[(\Delta W\widetilde{K} - E)(\Delta W\widetilde{K} - E)^\top\right] \tag{30}$$

$$= \operatorname{Tr}(\Delta W\widetilde{K}\widetilde{K}^\top\Delta W^\top) - 2\operatorname{Tr}(\Delta W\widetilde{K}E^\top) + \operatorname{Tr}(EE^\top). \tag{31}$$

Hence the full objective is

$$\begin{aligned} J(\Delta W) = \operatorname{Tr}(\Delta W\widetilde{K}\widetilde{K}^\top\Delta W^\top) - 2\operatorname{Tr}(\Delta W\widetilde{K}E^\top) + \operatorname{Tr}(EE^\top) \\ + \lambda \operatorname{Tr}(\Delta W C_r \Delta W^\top) + \epsilon\|\Delta W\|_F^2. \end{aligned} \tag{32}$$

Since C_r and $\widetilde{K}\widetilde{K}^\top$ are symmetric, taking the matrix derivative with respect to ΔW gives

$$\nabla_{\Delta W} J = 2\Delta W\widetilde{K}\widetilde{K}^\top - 2E\widetilde{K}^\top + 2\lambda\Delta W C_r + 2\epsilon\Delta W. \tag{33}$$

Setting the gradient to zero gives

$$\Delta W\widetilde{K}\widetilde{K}^\top + \lambda\Delta W C_r + \epsilon\Delta W = E\widetilde{K}^\top. \tag{34}$$

Equivalently,

$$\Delta W\left(\widetilde{K}\widetilde{K}^\top + \lambda C_r + \epsilon I\right) = E\widetilde{K}^\top. \tag{35}$$

The system matrix is positive definite for $\epsilon > 0$, so the normal equation gives

$$\Delta W = E\widetilde{K}^\top\left(\widetilde{K}\widetilde{K}^\top + \lambda C_r + \epsilon I\right)^{-1}. \tag{36}$$

Substituting $E = \widetilde{H}^\star - W\widetilde{K}$, we obtain

$$\boxed{\Delta W = \left(\widetilde{H}(O^\star) - W\widetilde{K}\right)\widetilde{K}^\top\left(\widetilde{K}\widetilde{K}^\top + \lambda C_r + \epsilon I\right)^{-1}}. \tag{37}$$

We use $\epsilon = 10^{-4}$. Without ϵ, $\widetilde{K}\widetilde{K}^\top + \lambda C_r$ has rank at most $r + 1 + n_r$ and is singular whenever this is below m. Let $S = \operatorname{span}\{\widetilde{K}, K_r\} \subseteq \mathbb{R}^m$. The system matrix maps S to itself and acts as ϵI on $S^\perp$, so $(\widetilde{K}\widetilde{K}^\top + \lambda C_r + \epsilon I)^{-1}\widetilde{K}$ has columns in S. Hence every row of ΔW lies in S, and $\Delta W k = 0$ for any key $k \perp S$: the edit acts only on directions spanned by the forget pseudo-keys and the retain keys.

Relation to MEMIT. Eq. (37) has the form of MEMIT's batch update $\Delta W = RK^\top(C_0 + KK^\top)^{-1}$ (Meng et al., 2023) under $K \leftarrow \widetilde{K}$, $R \leftarrow \widetilde{H}^\star - W\widetilde{K}$, and $C_0 \leftarrow \lambda C_r + \epsilon I$. MEMIT and AlphaEdit insert one key–value pair per fact with values optimized by gradient descent; WADE derives $r+1$ pseudo pairs from the moments of the forget-key and target distributions, so the number of constraints is set by covariance rank rather than forget-set size.

Surrogate descent. Let $\widehat{\mathcal{D}}_2(\Delta W) = \min_{O\in\mathcal{O}(r)} \|(W + \Delta W)\widetilde{K} - \widetilde{H}(O)\|_F^2$ be the truncated surrogate and $R(\Delta W) = \lambda \operatorname{Tr}(\Delta W C_r \Delta W^\top) + \epsilon\|\Delta W\|_F^2$.

Proposition 1 (Surrogate descent)**.** *Let $J(\Delta W) = \widehat{\mathcal{D}}_2(\Delta W) + R(\Delta W)$. The update in Eq.* (37) *satisfies $J(\Delta W) \le J(0)$.*

Proof. Let $U(\Delta W) = \|(W + \Delta W)\widetilde{K} - \widetilde{H}(O^\star)\|_F^2$. For every ΔW, $\widehat{\mathcal{D}}_2(\Delta W) \le U(\Delta W)$, since $O^\star$ is feasible for the inner minimization; at $\Delta W = 0$ equality holds because $O^\star$ is the minimizer for W. Hence $U + R$ majorizes J and touches it at 0. Since $\Delta W^\star$ in Eq. (37) minimizes $U + R$, $J(\Delta W^\star) \le U(\Delta W^\star) + R(\Delta W^\star) \le U(0) + R(0) = J(0)$. □

The update is thus one majorization–minimization step; alternating further O- and ΔW-updates would decrease J monotonically, but we use a single step.

D Additional MIA Results

Table 3 reports the aggregate membership-inference summary used in the main text. Here we break out each attack separately. AUC is computed with FORGET10 as members and HOLDOUT10 as non-members under the evaluator convention; values far from 0.5 in either direction indicate separability, so the leakage column reports $|\mathrm{AUC} - 0.5|$. KS-D and W1z compare each method's forget-example attack-score distribution to the retrain reference.

Table 6: Per-attack MIA results for the sequence-loss attack on TOFU (Llama-2-7B-Chat).

Method	AUC	$\lvert\mathrm{AUC} - 0.5\rvert$	KS-D↓	W1z↓
Retrain (reference)	0.392	0.108	0.000	0.000
Finetuned	0.999	0.499	0.995	2.132
NPO	0.098	0.402	0.520	1.600
LUNAR	0.613	0.113	0.235	0.433
WADE (ours)	0.530	**0.030**	**0.185**	0.537

Table 7: Per-attack MIA results for the Min-K attack on TOFU (Llama-2-7B-Chat).

Method	AUC	$\lvert\mathrm{AUC} - 0.5\rvert$	KS-D↓	W1z↓
Retrain (reference)	0.387	0.113	0.000	0.000
Finetuned	0.999	0.499	0.995	2.234
NPO	0.099	0.401	0.525	1.647
LUNAR	0.639	0.139	**0.118**	**0.199**
WADE (ours)	0.558	**0.058**	0.185	0.366

Table 8: Per-attack MIA results for the Min-K++ attack on TOFU (Llama-2-7B-Chat).

Method	AUC	$\lvert\mathrm{AUC} - 0.5\rvert$	KS-D↓	W1z↓
Retrain (reference)	0.502	0.002	0.000	0.000
Finetuned	0.999	0.499	0.993	1.424
NPO	0.090	0.410	0.665	68.490
LUNAR	0.867	0.367	0.818	1.218
WADE (ours)	0.787	**0.287**	**0.658**	**1.066**

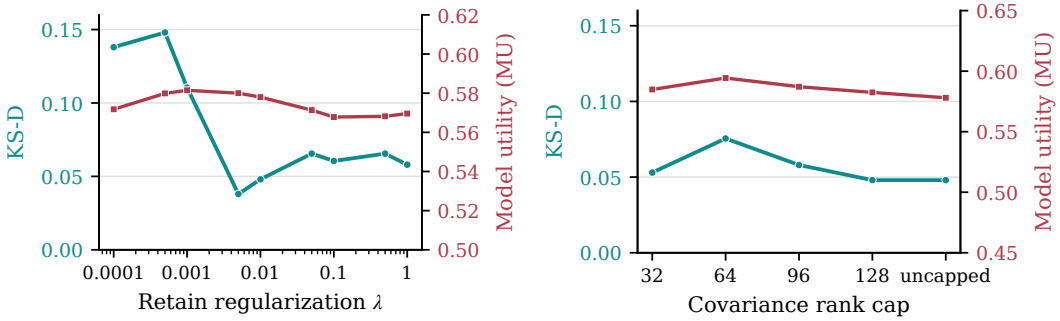
E Additional Hyperparameter Sensitivity

Figure 2 reports the sensitivity of WADE to the retain regularization coefficient λ and the covariance-rank cap on Llama-2-7B-Chat TOFU forget10. We report retrain-reference KS-D together with model utility (MU). For the retain regularization coefficient, the selected default lies in a low-KS-D region while maintaining utility. Very small values of λ give noticeably worse retrain-reference alignment, suggesting that the retain-key second moment term helps stabilize the closed-form edit rather than only preserving utility. Increasing λ beyond the selected region provides no clear utility gain and can slightly weaken alignment.

For covariance rank, the default WADE setting does not use a fixed manually chosen rank. Each edited layer selects adaptive PCA/SVD ranks by retaining the smallest number of covariance directions whose cumulative explained variance reaches 0.99, and the Procrustes step uses the common rank $r = \min(r_f, r_t)$. The rank-cap ablation imposes an additional maximum rank before this common-rank step. Thus, ranks 32, 64, 96, and 128 test sensitivity to restricting covariance alignment to fewer principal directions, while the uncapped setting corresponds to the default adaptive rank. In the Llama-2-7B-Chat layer 6–8 setting, the uncapped target covariance ranks are approximately 131, 104, and 78. The near-match between rank 128 and the uncapped setting is therefore mainly a sanity check, while the stricter caps show that moderate low-rank covariance alignment preserves much of WADE's retrain-reference alignment.

Table 9: Per-attack MIA results for the zlib-normalized loss attack on TOFU (Llama-2-7B-Chat).

Method	AUC	$\lvert\text{AUC} - 0.5\rvert$	KS-D↓	W1z↓
Retrain (reference)	0.318	0.182	0.000	0.000
Finetuned	1.000	0.500	0.995	2.384
NPO	0.137	0.363	0.463	2.593
LUNAR	0.566	0.066	**0.190**	0.764
WADE (ours)	0.478	**0.022**	0.205	**0.500**



(a) Retain regularization coefficient λ.

(b) Maximum covariance-rank cap.

Figure 2: Hyperparameter sensitivity of WADE on Llama-2-7B-Chat TOFU forget10. The selected λ lies in a low-KS-D region while preserving utility, and the rank-cap sweep shows that moderate low-rank covariance alignment preserves much of WADE's retrain-reference alignment.

F Additional Comparisons and Controls

Per-method sweeps. On Llama-2-7B-Chat forget10 we sweep WADE over $\lambda \in \{10^{-4}, 5{\times}10^{-4}, 10^{-3}, 5{\times}10^{-3}, 10^{-2}, 5{\times}10^{-2}, 0.1, 0.5, 1\}$ with layers 6–8 fixed; NPO over learning rate $\in \{10^{-5}, 2.5{\times}10^{-5}, 5{\times}10^{-5}\}$ $\times$ epochs $\in \{3, 5\}$; RMU over steering coefficient $\in \{0.5, 2, 8\}$ $\times$ layer $\in \{5, 7\}$; and LUNAR over learning rate $\in \{0.0025, 0.01, 0.04\}$ $\times$ layer $\in \{15, 19\}$. Every configuration is a complete run evaluated at its endpoint, without early stopping or intermediate-checkpoint selection, and the baseline grids receive a comparable or larger aggregate GPU budget than the nine WADE solves. Table 10(a) lists each method's non-dominated points. Pooled across methods, the frontier consists of WADE at $\lambda = 0.005$ and NPO; every RMU and LUNAR configuration is dominated. The default $\lambda = 0.01$ used in Table 1 is slightly dominated by $\lambda = 0.005$, since λ was fixed a priori rather than tuned on the evaluation split.

K-FADE. K-FADE has no author-released end-to-end TOFU recipe. We implemented its stated update with the public Kronfluence library (commit `db01b341e760f56f1a61e8f29d5a2fe486165a08`) on Llama-3.2-3B: 900 retain examples, Monte Carlo true-Fisher K-FAC over all gate/up/down projections in 28 layers (84 matrices), fp32 factors, a bf16 model, batch size 1, damping 10^{-8}, and step size $\alpha \in \{0.0025, 0.004, 0.006, 0.008, 0.011\}$, with the same split, retrain reference, and evaluator as WADE. Table 10(b) shows that the two non-dominated K-FADE points bracket WADE's default point; neither dominates it, and it dominates neither.

Capacity controls, paired regression, and edit order. Table 11 reports controls on Llama-2-7B-Chat forget10 with the same prompts, layers, and λ. The mean-only variant is the $r = 0$ special case of WADE, so its comparison with full WADE also changes the rank of the update; the norm-matched, covariance-only, and spectrum-matched variants separate update magnitude, rank, and spectrum from the target-specific covariance directions. Paired regression replaces the distributional term with $\sum_i \|W' k_f^i - h(x_i)\|_2^2$ over constructed forget–target pairs. This pairing is an artifact of construction: permuting the fictitious names across prompts leaves the intended unknown-state distribution unchanged but changes the regression targets, whereas WADE depends only on target moments and also applies to an unpaired target bank of a different size. Order controls run the same

Table 10: (a) Non-dominated (KS-D, MU) points of each method's sweep on Llama-2-7B-Chat TOFU forget10; † marks points on the pooled frontier. (b) K-FADE step-size sweep vs. WADE's default setting on Llama-3.2-3B.

(a) Per-method sweeps (Llama-2-7B-Chat)

Method	Configs	Non-dominated KS-D / MU
WADE	9	0.038 / 0.580†; 0.110 / 0.582
NPO	6	0.088 / 0.625†
RMU	6	0.135 / 0.586; 0.190 / 0.592; 0.358 / 0.601; 0.365 / 0.606
LUNAR	6	0.080 / 0.425; 0.133 / 0.433

(b) K-FADE (Llama-3.2-3B)

Method	KS-D↓	MU↑
WADE (default)	0.088	0.624
K-FADE α=0.0025	0.135	0.651
K-FADE α=0.004	0.073	0.619
K-FADE α=0.006	0.108	0.566
K-FADE α=0.008	0.143	0.481
K-FADE α=0.011	0.198	0.000

sequential solver with the layer schedule changed; a joint multi-layer solve is not an ordering control, since it changes the optimization problem and loses the layer-wise closed form.

Table 11: Controls on Llama-2-7B-Chat TOFU forget10. $\|\Delta W\|$: total update norm over edited layers. ‡Mean over 3 seeds.

Variant	KS-D↓	MU↑	$\|\Delta W\|$
WADE (mean + target covariance, ascending 6→8)	**0.048**	0.578	32.67
Mean only	0.230	0.223	7.12
Mean only, norm-matched	0.345	0.000	32.67
Covariance only	0.268	0.574	28.96
Same spectrum, random rotation‡	0.082	0.442	32.67
Shuffled eigenvalues‡	0.151	0.374	38.11
Paired regression	0.063	0.586	–
Edit order: descending 8→6	0.148	–	–
Edit order: random‡	0.113	–	–

Out-of-objective metrics. Table 12 reports three OpenUnlearning metrics that are not used to select any method: extraction strength (ES), exact memorization (EM), and ROUGE on forget answers, each as the absolute gap to retrain. WADE has the smallest ES gap and is on par with NPO on ROUGE, but trails NPO on EM; it improves on LUNAR on all three.

Table 12: Absolute gap to the retrain reference on memorization metrics (Llama-2-7B-Chat TOFU forget10; lower is better).

Method	ES gap↓	EM gap↓	ROUGE gap↓
NPO	0.025	**0.011**	0.091
LUNAR	0.020	0.153	0.284
WADE (ours)	**0.007**	0.066	**0.089**