CONFORMAL UNLEARNING: A NEW PARADIGM FOR OUTPUT CONTROL UNDER CONFORMAL PREDICTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Existing machine unlearning definitions generally constrain the objective of unlearning to approximating a model retrained from scratch after removing the forget set. However, this formulation equates unlearning to 'untraining' under all scenarios, presenting substantial practical limitations as it often neglects the gap between parameter-space proximity and output-level consistency, is vulnerable to falsification, and overstates retrained models as a universal gold standard for unlearning. To address these issues, we introduce conformal machine unlearning, a new paradigm that does not require a retrained model as the reference, and instead focuses on controlling the output of the model. We formalize conformal unlearning as an output-control framework that enforces high conformal coverage on retained data and high conformal miscoverage on forgotten data, develop empirical evaluation metrics, and present an algorithm that optimizes these conformal objectives. Extensive experiments on vision and text benchmarks verify that the proposed method effectively removes targeted information while preserving utility.

1 INTRODUCTION

The increasing deployment of machine learning systems in dynamic environments has underscored the need for models to adapt to evolving knowledge and semantics. In practice, entire categories of data may become obsolete, redefined, or classified as sensitive. Such scenarios necessitate the ability to efficiently remove the influence of outdated or irrelevant information from trained models, a challenge addressed by the field of *machine unlearning (MU)* (Cao & Yang, 2015; Bourtoule et al., 2021; Warnecke et al., 2023). MU aims to eliminate the impact of specific data points from a model without the computational overhead of retraining from scratch, thereby ensuring model reliability and compliance with evolving requirements (Cao & Yang, 2015). The need for selective forgetting is particularly pronounced in sensitive domains (Lu et al., 2022; Lambrou et al., 2011), where models must provide calibrated confidence while adapting to changes in guidelines, data removal requests, or outdated protocols. Despite these needs, the field of MU remains dominated by 'untraining' objectives (Triantafillou et al., 2026) which restrict the aim of an unlearning algorithm to producing a model as close as possible in parameters to a model retrained naively from scratch without access to the data to forget (Triantafillou et al., 2026). This restriction not only prevents the field from responding to flexible data removal needs but also confines the vision of MU practitioners.

Existing 'untraining'-based MU techniques fall into three categories: (i) data-structure-based partial retraining (Ginart et al., 2019; Bourtoule et al., 2021); (ii) gradient-influence and variational parameter updates (Warnecke et al., 2023; Graves et al., 2021; Nguyen et al., 2020); and (iii) knowledge-transfer and noise-based strategies (Chundawat et al., 2023a;b; Foster et al., 2025). Certified approaches bound residual information leakage via differential privacy (DP) (Dwork & Roth, 2014) or influence-function theory (Guo et al., 2020; Sekhari et al., 2021; Neel et al., 2021; Koh & Liang, 2017). However, these methods predominantly focus on 'untraining' by approximating a retrained (RT) model either directly or through a proxy objective (see Appendix A for a discussion of related work).

We identify three critical limitations faced by 'untraining'-based MU:

I1 Dependence on RT baselines for utility assessment: Conventional MU guarantees define unlearning as producing parameters indistinguishable from a model retrained without the forget data (Sekhari et al., 2021; Guo et al., 2020; Koloskova et al., 2025). However, this parameter-space

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.

focus obscures empirical evaluation: models that are parameter-close to a RT baseline may still exhibit divergent behavior on validation or held-out data (Bernstein et al., 2020). By the same token, models with exact output distributions as the RT baseline might diverge severely in their parameter distributions due to massive null spaces of the parameters. For instance, Table 1 shows accuracy deviations between the RT model and PABI (Koloskova et al., 2025), an (ϵ, δ)-certified indistinguishable method, on validation data. While these deviations are bounded by (ϵ, δ), they remain impractical to assess without referencing the RT model. Similarly, Fig. 2 illustrates perceptible differences in Grad-CAM overlays despite certified indistinguishability. This deems proximity in parameter space neither necessary nor sufficient for output certification. Moreover, such reliance on costly RT baselines for utility validation is infeasible at scale (Ginart et al., 2019; Warnecke et al., 2023; Neel et al., 2021).

I2 Forgeability in parameter-space unlearning: Parameter-space certification, as critiqued in (Thudi et al., 2022b), is susceptible to forgeability, where indistinguishable or identical parameter vectors can arise from different training datasets. This undermines the validity of unlearning definitions based solely on parameter similarity to a RT model. Certified approaches aiming for (ϵ, δ)-unlearning (Mu & Klabjan, 2025; Basaran et al., 2025; Chien et al., 2024; Koloskova et al., 2025; Guo et al., 2020; Ginart et al., 2019) are particularly prone to this issue, as indistinguishability in parameter space could be produced by training on differently curated datasets.

I3 RT models are not always the golden standard: Fig. 1 demonstrates that even when retraining leads to a significant reduction in accuracy on the training forget data, conformal prediction (CP) at a significance level of $\alpha = 0.05$ still achieves high coverage. This occurs despite the average size of the CP sets being relatively small (approximately 20 out of 100 labels). In other words, although the model's top-1 predictions are frequently incorrect on forget data (low accuracy), the true label often remains within the CP sets (high coverage), indicating that the model retains substantial knowledge about the distribution from which the forgotten data were drawn. This phenomenon, we term *fake conformal unlearning*, persists even when accuracy drops are pronounced (exceeding 40% across various numbers of forgotten clusters). It demonstrates that a RT model cannot be considered the golden standard in every unlearning scenario (Cooper et al., 2025; Shi et al., 2025), and this gap between what accuracy reports and what the prediction sets retain underscores the necessity for statistically rigorous definitions of unlearning that focus on output-control with clear and actionable control levels. This is not to say that targeting the RT model is an invalid objective for machine unlearning. Rather, we emphasize that while approximating the RT model, that is, 'untraining', is justified by the need to forget specific data points, unlearning categories or groups of data that share certain characteristics calls for a wider perspective on unlearning.

CP offers a natural resolution: by defining unlearning through coverage and miscoverage probabilities, which are directly observable and statistically quantifiable, we obtain objectives that are (i) independent of any baseline model, (ii) immune to forgeability concerns since they characterize prediction-set behavior rather than parameter values, and (iii) equipped with finite-sample guarantees under mild assumptions. Our main contributions are:

- We propose a conformal, probabilistic definition of unlearning that quantifies conformal forgetting directly, without reference to any retrain-from-scratch baseline.
- We introduce practical empirical metrics for evaluating uncertainty-aware conformal unlearning (Section 4).
- We develop a scalable unlearning algorithm that outputs an unlearned CP set, achieving strong forgetting of targeted data while preserving coverage on retained data.

Our approach shifts the unlearning objective from approximating the expensive RT model to a statistical *output-control* framework, where a successfully unlearned model is characterized by high uncertainty on the forgotten data and high confidence on the retained data. While prior work, such as (Shi et al., 2025), has made initial efforts to bridge traditional machine unlearning with CP, their approach does not fully address the unique challenges posed by conformal unlearning as defined in this work (cf. Appendix A).

2 PRELIMINARIES AND NOTATIONS

Let $X \in \mathcal{X}$ denote features and $Y \in \mathcal{Y}$ denote a label or response corresponding to X. We denote a dataset $\mathcal{D} \sim p$ if $\mathcal{D}$ consists of data points $(x, y) \in \mathcal{X} \times \mathcal{Y}$ generated independent and identically distributed (i.i.d.) from p.

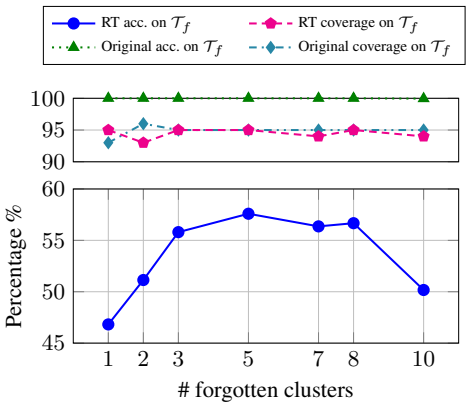
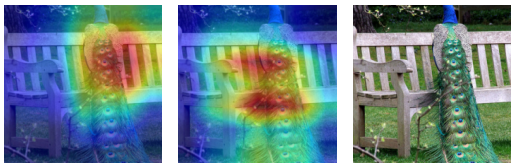


Figure 1: Accuracy (acc.) and coverage on the train forget data $\mathcal{T}_f$ for the RT and original models on CIFAR100.

Table 1: Validation forgotten/retained accuracy (%) for certified gradient clipping (PABI) and RT, forgetting 5 clusters in ImageNet100 (cf. Section 5).

Method	$\text{Acc}_{\text{val,for}}$	$\text{Acc}_{\text{val,ret}}$
Certified grad. clip. (PABI)	31.08	67.96
Retraining (RT)	34.79	71.25



(a) PABI (b) RT (c) Original

Figure 2: Grad-CAM overlays on a peacock image after cluster-wise forgetting. PABI yields a saliency map considerably different from that of RT, despite certified indistinguishability.

A machine learning model trained on the training set $\mathcal{D}_{\text{train}} \sim p_{\text{data}}$ produces a model f_{θ_o} with parameters θ_o. Let $\mathbb{P}_{\text{data}}$ denote the probability measure corresponding to p_{data}.

We focus on *conformal unlearning*: Let a target variable $W \in \mathcal{W}$ encode the characteristics to be forgotten. Our objective is to forget the influence of data that are generated conditioned on $W \in \mathcal{W}_{\text{forget}}$, where $\mathcal{W}_{\text{forget}} \subset \mathcal{W}$ defines the forget criteria. Let $\pi_f = \mathbb{P}(W \in \mathcal{W}_{\text{forget}}) = 1 - \pi_r$ so that

$$p_{\text{data}} = \pi_r p_r + \pi_f p_f, \tag{1}$$

where the retain and forget sets are drawn from the conditional probability density functions (pdfs):

$$\begin{aligned} p_r(\cdot, \cdot) &= p_{X,Y|W}(\cdot, \cdot \mid W \in \mathcal{W} \backslash \mathcal{W}_{\text{forget}}), \\ p_f(\cdot, \cdot) &= p_{X,Y|W}(\cdot, \cdot \mid W \in \mathcal{W}_{\text{forget}}), \end{aligned} \tag{2}$$

respectively. We set $\mathcal{D}_{\text{unlearn}} = \mathcal{D}_f \cup \mathcal{D}_r \sim p_{\text{data}}$, which is used for empirical unlearning and constitutes forget points ($\mathcal{D}_f$) distributed according to p_f and retain points ($\mathcal{D}_r$) distributed according to p_r. Note that we do not require that $\mathcal{D}_{\text{unlearn}} \subset \mathcal{D}_{\text{train}}$ although this is common in practice.

An important assumption used throughout this work is that $p_f \neq p_r$, i.e., the distributions are distinguishable. Examples of conformal unlearning include class-level forgetting ($W = Y$) (Tarun et al., 2024; Chundawat et al., 2023a) and feature-, subspace-, or semantic-based criteria ($W = X$, $W = \Pi X$, or $W = h(X)$, where Π is a subspace projector and h is a feature transformation map) (Warnecke et al., 2023).

In the traditional MU literature, a MU algorithm $\mathfrak{U}$ transforms the model parameters θ_o into θ_u by utilizing $\mathcal{D}_{\text{unlearn}}$ and possibly other information so that the unlearned model f_{θ_u} approximates the RT model. In conformal unlearning, as defined in Section 3, we depart from this perspective and instead focus on directly quantifying the forgetting and retaining performance of f_{θ_u} on p_f and p_r, respectively, in an output-control manner without reference to the RT model.

Split CP The split CP framework (Angelopoulos & Bates, 2023; Shafer & Vovk, 2008; Foygel Barber et al., 2023) provides distribution-free prediction sets with guaranteed coverage. For a trained model f_θ, a nonconformity score $s(X, Y; \theta)$ is a measure of the discrepancy of f_θ prediction

over X, and smaller values indicate better conformity. In classification tasks, a common choice is $s(X, Y; \theta) \triangleq 1 - p_\theta(Y \mid X)$, where p_θ represents softmax probability (Sadinle et al., 2019). The regression case is treated in Appendix C.

In split CP, the dataset is divided into a training set $\mathcal{D}_{\text{train}}$ for training the model f_θ, a calibration set $\mathcal{D}_{\text{calib}}$ of size n, and a test set $\mathcal{D}_{\text{test}}$. Given a significance level α, let $\hat{q}_\alpha$ represent the $(1-\alpha)(1+1/n)$ quantile of the scores computed from $\mathcal{D}_{\text{calib}}$. For any test point $(X, Y) \in \mathcal{D}_{\text{test}}$, construct the prediction set

$$\mathcal{C}_{\theta,\hat{q}_\alpha}(X) = \{y : s(X, y; \theta) \leq \hat{q}_\alpha\}. \tag{3}$$

By assuming that $\mathcal{D}_{\text{calib}}$ and $\mathcal{D}_{\text{test}}$ are exchangeable (Angelopoulos & Bates, 2023; Shafer & Vovk, 2008; Foygel Barber et al., 2023), the following *coverage* is achieved:

$$\mathbb{P}(Y \in \mathcal{C}_{\theta,\hat{q}_\alpha}(X)) = \mathbb{P}(s(X, Y; \theta) \leq \hat{q}_\alpha) \geq 1 - \alpha, \tag{4}$$

where the probability accounts for the randomness in both the calibration set and the test point. The quantile-based CP procedure described above is denoted as $\mathfrak{CP}(\theta, s)$, which outputs the prediction set $\mathcal{C}_\theta \triangleq \mathcal{C}_{\theta,\hat{q}_\alpha}$. The guarantee in (4) holds for any θ, as it is induced by the calibration procedure rather than the specific model parameters. Our MU objective (detailed in Section 3) leverages this property, under the exchangeability condition stated in Assumption 1, to achieve conformal unlearning objectives.

3 Conformal Machine Unlearning

3.1 Definitions and Theory

Consider a machine learning model f_{θ_o} and a quantile-based CP procedure $\mathfrak{CP}(\theta, s)$, where the probability measure in (4) is $\mathbb{P} = \mathbb{P}_{\text{data}}$. For convenience, we use the notation $\mathbb{P}$ in place of $\mathbb{P}_{\text{data}}$ throughout the rest of this paper whenever there is no confusion.

Let $\mathbb{P}_r$ denote the probability measure under which $(X, Y) \sim p_r$ and $\mathcal{D}_{\text{calib}} \sim p_{\text{data}}$. Similarly, let $\mathbb{P}_f$ denote the probability measure under which $(X, Y) \sim p_f$ and $\mathcal{D}_{\text{calib}} \sim p_{\text{data}}$.

Definition 1 ((α,β)-conformal unlearning)**.** *For $0 \leq \alpha \leq \beta \leq 1$, an unlearning algorithm $\mathfrak{U}$ is said to be (α,β)-conformal unlearning if the prediction sets $\mathcal{C}_{\theta_u}(X)$ produced by $\mathfrak{CP}(\theta_u, s)$ satisfy:*

$$\mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) \geq 1 - \alpha, \tag{5}$$

$$\mathbb{P}_f(Y \notin \mathcal{C}_{\theta_u}(X)) \geq \beta. \tag{6}$$

Definition 1 places uncertainty sets at the core of unlearning, moving beyond point-estimate objectives such as misclassification (Graves et al., 2021; Hsu et al., 2025). Intuitively, a high miscoverage indicates that the unlearned model demonstrates significant uncertainty about p_f while maintaining confidence on p_r. Here, α represents the user-specified miscoverage rate inherent to CP, which varies depending on the application. Meanwhile, in sensitive applications, such as unlearning confidential or regulated information, achieving a large β is crucial to minimize the risk of information leakage. In less critical contexts, a lower β may suffice; for instance, if the goal is to forget a broad category, occasional inclusion in a prediction set may not be consequential. The coverage and miscoverage objectives in (5) and (6) are independent of the underlying data distribution and the model's parameter space. Consequently, if the original model prior to unlearning already achieves uniformly high miscoverage rates on the forget data, no unlearning is required, as the model already exhibits significant uncertainty on that subset. In other words, without restricting the unlearning target to a specific parameter set, any model that satisfies the coverage and miscoverage objectives is deemed successful in unlearning the specified forget set. This ensures that forgeability (Thudi et al., 2022b) is *not* a concern under conformal unlearning.

To analyze the properties of conformal unlearning, we make the following fundamental assumption regarding the exchangeability of data points in $\mathcal{D}_{\text{calib}}$ and the *test* forget set $\mathcal{V}_f$ and retain set $\mathcal{V}_r$. All proofs are deferred to Appendix G.

Assumption 1. *The data points in $\mathcal{D}_{\text{calib}}$ and $\mathcal{D}_{\text{test}} = \mathcal{V}_f \cup \mathcal{V}_r$ are exchangeable conditioned on the model f_{θ_u}.*

Lemma 1. *Suppose Assumption 1 holds and $\pi_r > 0$. Then,* (6) *implies* (5) *for $\beta \geq \alpha$.*

On the other hand, suppose (5) holds. In the context of conformal unlearning, if the conformity scores of the data samples in $\mathcal{D}_{\text{calib}}$ are continuous and free of ties (Lei et al., 2018) (a condition that can always be met by adding a small random perturbation), then letting $n = |\mathcal{D}_{\text{calib}}|$, $\widetilde{\beta} = \mathbb{P}_f(Y \notin \mathcal{C}_{\theta_u}(X))$, and assuming $\pi_f > 0$, we have:

$$1 - \alpha + \frac{1}{n+1} \geq \mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) \geq (1-\alpha)(1-\pi_f) + (1-\widetilde{\beta})\pi_f \implies \widetilde{\beta} \geq \alpha - \frac{1}{(n+1)\pi_f}.$$

This inequality shows that (6) cannot be guaranteed for $\beta \geq \alpha$ without a non-trivial unlearning algorithm. It highlights the necessity of designing effective MU algorithms to meet the requirements of Definition 1. Furthermore, it suggests that β cannot be arbitrarily high. CP imposes constraints on the maximum achievable β, which depends on the likelihood of the forget set $\mathcal{V}_f$.

Lemma 2. *Suppose Assumption 1 and* (6) *hold. Then,*

$$\pi_f \beta \leq \alpha \leq \beta. \tag{7}$$

Although this bound uses the forgotten cohort probability, in practical applications practitioners do not typically forget large portions of the data such that the bound forces β to be small. In general, π_f is practically up to 5% or smaller. If π_f becomes so high, then one would better consider their initial training to be flawed.

Any (α, β)-conformal unlearning algorithm with relatively small α and large β has a statistically strong unlearning property. For a calibration set $\mathcal{D}_{\text{calib}}$ exchangeable with $\mathcal{D}_{\text{test}}$, the cutoff (α, β)-conformal unlearning method operating on $\mathcal{D}_{\text{calib}}$ is given by $\alpha = \frac{1}{n+1}$ since any smaller value of α leads to $\hat{q}_\alpha = \infty$ and $\mathcal{C}_{\theta_u}(X) = \mathcal{Y}$ for all X. Hence, the unlearning collapses, and all samples become covered with the trivial set. In that case, the forget set miscoverage probability in (6) is 0, and is excluded from Definition 1. In another trivial case where $\alpha = 1$, Lemma 2 gives $\beta = 1$, which is expected. Note that not all possible pairs (α, β) are achievable, depending on the given CP procedure. Exploring the feasibility region for coverage and miscoverage pairs presents an interesting direction for future research. In Theorem C.1, we show an (α,1)-conformal unlearning algorithm under Assumption C.1 in the regression case, testifying to the achievability of conformal unlearning bounds. Although Theorem C.1 establishes theoretical conditions for achieving perfect conformal unlearning ($\beta = 1$) in the regression example, these conditions may not always hold in practical scenarios. Consequently, empirical methodologies are essential to effectively optimize the unlearning process, a need for which Section 4 introduces a specific conformal unlearning framework.

Assumption 1 is critical for ensuring the validity of the coverage guarantees in (4), (5), and (6). In practical scenarios, exchangeability can often be achieved in class-wise or group-wise unlearning by reserving validation points for each class or group during the training process. If exchangeability is violated, more general CP frameworks, such as the non-exchangeable CP framework with coverage gap correction (Foygel Barber et al., 2023), can be employed. In such cases, the coverage gap must be explicitly incorporated into all relevant bounds, including (4), (5), and (6). A brief discussion of the non-exchangeable setting is provided in Appendix F. Moreover, the important case of using a different non-conformity scoring function between the unlearner and the user is extensively discussed under Appendix E.

3.2 Zero-Shot Conformal Unlearning

In the case of *zero-shot* MU (as defined in (Foster et al., 2025)) where sampling from the forget distribution p_f is not available, we are restricted to a calibration set whose samples are from the retained distribution p_r (i.e., $\mathcal{D}_{\text{calib}} \sim p_r$).

Proposition 1. *Suppose $\mathcal{D}_{\text{calib}}$ and $\mathcal{V}_r$ are exchangeable, and are independent of $\mathcal{V}_f$. Let $(X_f, Y_f) \sim p_f$, and $(X_r, Y_r) \sim p_r$. An (α,β)-conformal unlearning algorithm $\mathfrak{U}$ based on $\mathcal{D}_{\text{calib}}$ and $\mathcal{V}_r$ yields*

$$\mathbb{P}(s(X_f, Y_f; \theta_u) \geq s(X_r, Y_r; \theta_u)) \geq \beta(1-\alpha). \tag{8}$$

Proposition 1 establishes that for small α and large β, an (α,β)-conformal unlearning algorithm demonstrates, on average, higher non-conformity on the forget data compared to the retained data. When the conformity scoring function s corresponds to a loss function, this implies that the loss incurred on $\mathcal{V}_f$ is, with high probability, greater than the loss on $\mathcal{V}_r$. This result aligns with the intuition articulated following Definition 1.

4 EMPIRICAL METRICS AND OPTIMIZATION FRAMEWORK

We introduce two metrics as practical measures for evaluating coverage and miscoverage. Furthermore, we present the Excluding Forgotten data by Adjusting CoveragE (EFACE) framework, a systematic approach designed to optimize these metrics and achieve effective unlearning.

4.1 EMPIRICAL CONFORMAL UNLEARNING METRICS

Excessively large CP sets tend to lose their informativeness, which is undesirable for predictions on p_r but aligns with the objective of unlearning on p_f. Specifically, for a retained dataset $\mathcal{V}_r$ and a forget dataset $\mathcal{V}_f$ and positive integers c, d, the *Coverage Frequency* (CF) at set-level c and the *Miscoverage Frequency* (mCF) at set-level d for a prediction set $\mathcal{C}(\cdot)$ are defined as:

$$\mathsf{C}_c(\mathcal{V}_r) = \frac{1}{|\mathcal{D}_{r,c}|} \sum_{(x,y)\in\mathcal{D}_{r,c}} \mathbf{1}\{y \in \mathcal{C}(X)\}, \tag{9}$$

$$\mathsf{M}_d(\mathcal{V}_f) = \frac{1}{|\mathcal{D}_{f,d}|} \sum_{(x,y)\in\mathcal{D}_{f,d}} \mathbf{1}\{y \notin \mathcal{C}(X)\}, \tag{10}$$

where $\mathcal{D}_{r,c} = \{(x,y) \in \mathcal{V}_r : |\mathcal{C}(X)| \le c\}$, $\mathcal{D}_{f,d} = \{(x,y) \in \mathcal{V}_f : |\mathcal{C}(X)| \le d\}$, and $\mathbf{1}\{\cdot\}$ is the indicator function.

The thresholds c (for retained data) and d (for forgotten data) define the maximum prediction-set sizes considered *informative* by the unlearning framework. Prediction sets exceeding these thresholds are classified as "inefficient", as they are excessively broad and indicative of low model confidence (Shafer & Vovk, 2008). The selection of c and d is context-dependent and reflects the unlearner's tolerance for uncertainty in the prediction sets (see Appendix J.3 for an illustrative example). Notably, when $c = d = |\mathcal{Y}|$, CF and mCF simplify to the standard coverage and miscoverage frequencies respectively. A model that satisfies Definition 1 is expected to have bounded coverage and miscoverage probabilities at set-levels c and d, respectively, up to the corrections below.

Corollary 1. *Under the same assumptions as Lemma 1, suppose a MU algorithm $\mathfrak{U}$ is (α,β)-conformal unlearning. For $c, d \in \{0,\dots,|\mathcal{Y}|\}$, assume $\mathbb{P}_r(|\mathcal{C}_{\theta_u}(X)| > c) \le \zeta_c$, and $\mathbb{P}_f(|\mathcal{C}_{\theta_u}(X)| > d) \le \eta_d$. Then, we have*

$$\mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X) \,|\, |\mathcal{C}_{\theta_u}(X)| \le c) \ge 1 - \alpha - \zeta_c, \tag{11}$$

$$\mathbb{P}_f(Y \notin \mathcal{C}_{\theta_u}(X) \,|\, |\mathcal{C}_{\theta_u}(X)| \le d) \ge \beta - \eta_d. \tag{12}$$

4.2 UNLEARNING VIA OPTIMIZING CONFORMAL SETS

For a CP set $\mathcal{C}_{\theta,t}$ and an unlearning dataset $\mathcal{N} \in \{\mathcal{D}_r, \mathcal{D}_f\}$, the miscoverage loss is defined as follows:

$$L_{\mathrm{cov},\mathcal{N}}(\mathcal{C}_{\theta,t}) \triangleq \mathbb{P}_i(Y \notin \mathcal{C}_{\theta,t}(X)); \quad i \in \{r, f\}. \tag{13}$$

Its corresponding empirical version is given by

$$\widehat{L}_{\mathrm{cov},\mathcal{N}}(\mathcal{C}_{\theta,t}) \triangleq \frac{1}{|\mathcal{N}|} \sum_{(x,y)\in\mathcal{N}} \mathbf{1}\{y \notin \mathcal{C}_{\theta,t}(x)\}. \tag{14}$$

We recast the conformal unlearning problem as an empirical risk minimization (ERM) task. The primary objective is to maximize the miscoverage on the unlearning forget set $\mathcal{D}_f$ while simultaneously minimizing the miscoverage on the unlearning retain set $\mathcal{D}_r$, thereby adhering to the principles of conformal unlearning. The resulting optimization problem is formulated as follows:

$$\min_{\theta,t} \widehat{L}_{\mathrm{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) - \widehat{L}_{\mathrm{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}). \tag{15}$$

Note that (15) targets the coverage and miscoverage levels of Definition 1 directly and leaves the prediction-set sizes unconstrained, so the set-levels c and d enter only through the CF and mCF metrics used for evaluation; we report an ablation that additionally constrains the set sizes during unlearning in Appendix J. In downstream applications, decision-makers may choose to reconformalize t to re-establish marginal conformal validity across the data mixture. We give generalization bounds for (15) in Appendix D.1. The pseudo-code of EFACE is presented in Algorithm 1. We address practical relaxations to make the problem differentiable in Appendix D.

5 NUMERICAL EXPERIMENTS

Datasets and Models. In the main text, we evaluate on ImageNet100 (Shekhar, 2021), a 100-class subset of ImageNet, with a pretrained RepVGG-A2 model, and we defer CIFAR100 (Krizhevsky, 2009) and 20NewsGroups (Lang, 1995) to Appendix I.2. Unless otherwise specified, we average over six random seeds, except for RT and PABI, which we average over three seeds due to their significant computational overhead; in all cases, the variability across seeds is minimal.

Data Partitions. We use training forget/retain splits $(\mathcal{T}_f, \mathcal{T}_r)$, unlearning forget/retain splits $(\mathcal{D}_f, \mathcal{D}_r)$, unseen forget/retain splits $(\mathcal{V}_f, \mathcal{V}_r)$, and a calibration set $\mathcal{D}_{\text{calib}}$, all of which are disjoint. EFACE optimizes over $\mathcal{D}_f$ and $\mathcal{D}_r$, so when these sets are disjoint from the training data we call the scenario *out-sample unlearning* (*Out*), and when $\mathcal{D}_f \subset \mathcal{T}_f$ and $\mathcal{D}_r \subset \mathcal{T}_r$ we call it *in-sample unlearning* (*In*). In all experiments, we reconformalize every method on $\mathcal{D}_{\text{calib}}$ once unlearning is complete (cf. Algorithm 1). Therefore, the distribution-free guarantee of (4) is claimed only for the held-out data $\mathcal{V}_r$ and $\mathcal{V}_f$, which are exchangeable with $\mathcal{D}_{\text{calib}}$, whereas the numbers on $\mathcal{T}_r$, $\mathcal{T}_f$, $\mathcal{D}_r$, and $\mathcal{D}_f$ are in-sample diagnostics, reported to show that even points used in training or unlearning are no longer covered.

In cluster-wise unlearning, we partition the training data into $k = |\mathcal{Y}|$ clusters in the embedding space using k-means and assign each point the pseudo-label of its nearest centroid, so the forget data, selected by pseudo-label, share common high-level characteristics as required by conformal unlearning (cf. Section 2). Full details are given in Appendix I.1.

Baselines. We compare against $\nabla\tau$ (Trippa et al., 2024), AMN (Graves et al., 2021), SCRUB (Kurmanji et al., 2023), BADT (Chundawat et al., 2023a), SSD (Foster et al., 2024), UNSIR (Tarun et al., 2024), SalUn (Fan et al., 2024), Boundary Shrink (BS) and Boundary Expanding (BE) (Chen et al., 2023), and CPU (Shi et al., 2025), which together span several unlearning approaches, and we explain these methods and why they are chosen in Appendix H.2. We also compare against the RT baseline and the certified unlearning method PABI (Koloskova et al., 2025), both applied to $\mathcal{T}_r$, which, to our knowledge, is the only certified method applicable to our setup.

Evaluation Metrics. We report $\mathsf{C}_{\mathcal{D}}(c)$ on the retained subsets $(\mathcal{D}_r, \mathcal{T}_r, \mathcal{V}_r)$ and $\mathsf{M}_{\mathcal{D}}(d)$ on the forgotten subsets $(\mathcal{D}_f, \mathcal{T}_f, \mathcal{V}_f)$ with $c = d$, together with the harmonic mean $H = 6/\sum_i x_i^{-1}$ of these six conformal metrics x_i (with $H = 0$ if any $x_i = 0$), and $A_{\mathcal{D}}$ denotes the accuracy on a subset $\mathcal{D}$. For RT and PABI, which unlearn on the entire retained dataset $\mathcal{T}_r$, we omit the results on $\mathcal{D}_r \subset \mathcal{T}_r$ and $\mathcal{D}_f \subset \mathcal{T}_f$. Detailed information to ensure reproducibility is provided in Appendix I.1.

The main text presents cluster-wise unlearning on ImageNet100, and Appendix I.2 presents the comprehensive results for the vision and text datasets in both the cluster-wise and class-wise cases.

5.1 RESULTS AND DISCUSSION

5.1.1 COVERAGE AND MISCOVERAGE FREQUENCIES

In Table 2, the best, second, and third best scores are highlighted in **red**, **blue**, and **cyan**, respectively. Results from the original (OR) model (prior to unlearning) are included for reference. We use green to highlight results that appear favorable at first glance, and gray to flag corresponding results from the same method—sometimes in a different table or scenario—that reveal poor performance or expose the green-highlighted result as misleading. When a method shows only green-highlighted results with no gray counterpart, the highlighting simply indicates genuinely strong performance worthy of note.

From Table 2, we observe that all methods achieve high retained coverage above the $1 - \alpha = 0.95$ threshold. However, $\nabla\tau$ fails to cover the retained data or miscover the forget data in the in-sample case. The reason is that the quantile of $\mathcal{D}_{\text{calib}}$ becomes $\hat{q}_\alpha = 1.00$, which renders all prediction sets to be full size (trivial sets). When we set $c = d = 50 < 100 = |\mathcal{Y}|$, there are no points with such set sizes, and hence the coverage and miscoverage frequencies are both 0.

BADT exhibits limited effectiveness in achieving high miscoverage levels on the forget subsets. This method relies on the Kullback-Leibler (KL) divergence for unlearning, which appears insufficient in the context of cluster-wise forgetting. The semantic overlap between forget and retain points in the feature space complicates the enforcement of distinct KL divergence values between these groups,

Table 2: ImageNet100, RepVGG-a2 *cluster*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten clusters. Coverage/miscoverage results.

Split	Method	$\mathsf{C}_c(\mathcal{D}_r) \uparrow$	$\mathsf{M}_d(\mathcal{D}_f) \uparrow$	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$
	OR	0.99 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.02 ± 0.00	0.01 ± 0.00
In	$\nabla\tau$	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	AMN	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00	0.24 ± 0.02	0.98 ± 0.00	0.21 ± 0.02	0.46 ± 0.02
	SalUn	1.00 ± 0.00	0.54 ± 0.24	0.98 ± 0.00	0.11 ± 0.02	0.97 ± 0.00	0.18 ± 0.04	0.31 ± 0.03
	SCRUB	1.00 ± 0.00	0.33 ± 0.09	1.00 ± 0.00	0.32 ± 0.09	0.99 ± 0.00	0.35 ± 0.08	0.50 ± 0.06
	BADT	0.99 ± 0.00	0.10 ± 0.01	0.99 ± 0.00	0.10 ± 0.01	0.98 ± 0.00	0.17 ± 0.02	0.21 ± 0.01
	SSD	0.99 ± 0.00	0.04 ± 0.04	0.99 ± 0.00	0.05 ± 0.05	0.98 ± 0.00	0.07 ± 0.05	0.10 ± 0.05
	BS	1.00 ± 0.00	0.95 ± 0.00	0.99 ± 0.00	0.22 ± 0.01	0.98 ± 0.00	0.21 ± 0.01	0.45 ± 0.01
	BE	1.00 ± 0.00	0.56 ± 0.04	0.99 ± 0.00	0.44 ± 0.04	0.99 ± 0.00	0.48 ± 0.04	0.66 ± 0.02
	CPU	1.00 ± 0.00	0.02 ± 0.00	0.98 ± 0.00	0.02 ± 0.00	0.97 ± 0.00	0.03 ± 0.00	0.04 ± 0.00
	EFACE	1.00 ± 0.00	0.85 ± 0.02	1.00 ± 0.00	0.78 ± 0.01	1.00 ± 0.00	0.79 ± 0.00	0.89 ± 0.00
Out	$\nabla\tau$	1.00 ± 0.00	0.02 ± 0.02	0.99 ± 0.00	0.01 ± 0.00	0.98 ± 0.00	0.02 ± 0.00	0.03 ± 0.01
	AMN	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00	0.12 ± 0.02	0.98 ± 0.00	0.14 ± 0.02	0.31 ± 0.03
	SalUn	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.29 ± 0.04	0.98 ± 0.00	0.33 ± 0.05	0.00 ± 0.00
	SCRUB	1.00 ± 0.00	0.99 ± 0.01	1.00 ± 0.00	0.56 ± 0.01	0.99 ± 0.00	0.56 ± 0.02	0.79 ± 0.01
	BADT	0.95 ± 0.00	0.03 ± 0.01	1.00 ± 0.00	0.01 ± 0.00	0.98 ± 0.00	0.05 ± 0.00	0.05 ± 0.00
	SSD	0.96 ± 0.00	0.06 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.02 ± 0.00	0.02 ± 0.00
	BS	0.98 ± 0.00	0.96 ± 0.00	0.99 ± 0.00	0.31 ± 0.00	0.99 ± 0.00	0.37 ± 0.00	0.60 ± 0.00
	BE	0.98 ± 0.00	0.81 ± 0.03	0.99 ± 0.00	0.28 ± 0.01	0.99 ± 0.00	0.30 ± 0.02	0.53 ± 0.01
	CPU	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.01 ± 0.00	0.97 ± 0.00	0.03 ± 0.00	0.00 ± 0.00
	EFACE	0.99 ± 0.00	0.97 ± 0.02	1.00 ± 0.00	0.58 ± 0.01	0.99 ± 0.00	0.59 ± 0.01	0.80 ± 0.01

thereby limiting the method's efficacy. In contrast, SCRUB incorporates an additional fine-tuning cross-entropy term over the retained data, supplementing the KL divergence-based objectives. This enhancement improves its performance relative to BADT, with out-sample results surpassing in-sample results—likely due to the KL divergence's greater effectiveness in distinguishing unseen data from training data. SSD, however, fails to demonstrate significant forgetting in both in-sample and out-sample scenarios, with miscoverage levels converging to $\beta \approx \alpha$ despite extensive hyperparameter tuning. This outcome may stem from the similarity of parameter importance scores between forget and retain data in the cluster forgetting case, which undermines the method's ability to differentiate between the two. AMN achieves near-perfect miscoverage on $\mathcal{D}_f$ (≈ 1.00), but this performance does not generalize to $\mathcal{T}_f$ (≈ 0.12) or $\mathcal{V}_f$ (≈ 0.14). This overfitting behavior persists even when in-sample data are utilized for unlearning. Nevertheless, AMN consistently maintains retained coverage above the target threshold of $1 - \alpha = 0.95$. Similarly, BS and BE miscover $\mathcal{D}_f$ well but generalize less to $\mathcal{T}_f$ and $\mathcal{V}_f$, while SalUn drives the accuracy on $\mathcal{D}_f$ to almost zero yet keeps the forget data covered by near-full prediction sets, an instance of fake conformal unlearning.

In contrast, EFACE consistently satisfies coverage above 0.95 on the retained subsets and high miscoverage on the forget subsets, leading in H in both scenarios, by a margin of 0.23 over the next best method in the in-sample case and on par with the strongest baseline out-sample. Moreover, it demonstrates consistent generalizability from $\mathcal{D}_f$ to $\mathcal{T}_f$ and $\mathcal{V}_f$, which is where the other methods lose most of their miscoverage.

Table 3: ImageNet100, ResNet18 *cluster*-wise forgetting with $c = d = 100$, $\alpha = 0.05$, and 5 forgotten clusters. Coverage/miscoverage, accuracy (%) and unlearning time (seconds) results.

Split	Method	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$	$A(\mathcal{T}_r) \uparrow$	$A(\mathcal{T}_f) \downarrow$	$A(\mathcal{V}_r) \uparrow$	$A(\mathcal{V}_f) \downarrow$	Tsec↓
	RT	1.00 ± 0.00	0.09 ± 0.01	0.96 ± 0.00	0.09 ± 0.01	0.19 ± 0.03	99.54 ± 0.00	36.14 ± 1.05	71.25 ± 0.14	34.79 ± 1.23	22814.00 ± 66.73
	PABI	1.00 ± 0.00	0.08 ± 0.01	0.96 ± 0.00	0.08 ± 0.01	0.15 ± 0.01	99.54 ± 0.00	35.34 ± 1.61	67.96 ± 0.34	31.08 ± 0.82	11463.52 ± 3.90
In	EFACE	1.00 ± 0.00	0.29 ± 0.02	0.97 ± 0.00	0.33 ± 0.01	0.56 ± 0.01	97.60 ± 0.09	6.69 ± 0.63	64.94 ± 0.42	3.12 ± 0.63	3175.85 ± 14.55
Out	EFACE	1.00 ± 0.00	0.33 ± 0.03	0.97 ± 0.00	0.37 ± 0.03	0.61 ± 0.02	94.50 ± 0.58	20.50 ± 3.07	60.22 ± 0.83	11.80 ± 1.67	410.35 ± 5.25

5.1.2 FAKE CONFORMAL UNLEARNING

Table 6 in Appendix I.2 illustrates the phenomenon of *fake conformal unlearning* (cf. Section 1), where several methods exhibit substantial accuracy degradation on the training-forgotten split $\mathcal{T}_f$—for

instance, $\nabla\tau$ (over 60%), AMN (over 60%), and BADT (over 50%)—yet fail to achieve the desired miscoverage rates on the same data, remaining below the target significance level α. This discrepancy indicates that, despite the observed accuracy drop, the conformal prediction sets $\mathcal{C}_{\theta_u}(X)$ frequently include the true label, even when constrained to small prediction set sizes ($c \leq 50$). Consequently, these samples are *covered* rather than *miscovered*, undermining the objective of effective unlearning.

The root cause of this inconsistency lies in the fundamentally different objectives of accuracy and conformal coverage. Accuracy penalizes any top-1 prediction error, whereas conformal coverage only requires the true label to be included within the prediction set, irrespective of its rank. As a result, methods that merely expand prediction sets—or fail to sufficiently reduce their size—can exhibit significant accuracy degradation without achieving genuine forgetting. For example, while the accuracy drop on $\mathcal{T}_f$ is comparable between $\nabla\tau$ and EFACE, their mCF rates differ markedly (0.00 vs. 0.78, respectively; cf. Table 2). Notably, in Table 3, both the RT model and the certified unlearning method (PABI) also exhibit signs of fake conformal unlearning, as evidenced by their negligible miscoverage rates on forget data ($\beta \approx \alpha = 0.05$). These findings underscore the importance of adopting global, coverage-based criteria to enable uncertainty-aware evaluation.

These results reinforce the argument presented in Section 1: **accuracy alone is an insufficient metric for evaluating effective conformal unlearning** and may instead serve as an indicator of *fake conformal unlearning*. In contrast, EFACE demonstrates a consistent alignment between reductions in accuracy on forget data and corresponding increases in miscoverage, ensuring that true labels are systematically excluded from $\mathcal{C}_{\theta_u}(X)$ at the specified prediction set size. Simultaneously, EFACE maintains high retained coverage ($\geq 1 - \alpha$), thereby satisfying the requirements of Definition 1. This alignment between coverage and miscoverage highlights that EFACE does not indiscriminately degrade logits but instead strategically adjusts prediction sets to ensure that forgotten concepts are effectively unsupported, while retained concepts remain reliably covered.

When unlearning is performed using proxy out-sample data $\mathcal{D}_f$, EFACE effectively reduces coverage on $\mathcal{T}_f$, as expected in the context of conformal unlearning. Since conformal unlearning targets data with shared characteristics, $\mathcal{D}_f$ and $\mathcal{T}_f$ are likely to occupy similar regions in the feature space. Consequently, shifting decision boundaries to miscover $\mathcal{D}_f$ naturally impacts $\mathcal{T}_f$. The observed increase in miscoverage, coupled with the corresponding drop in accuracy on both forgotten splits, demonstrates the intended effect of conformal unlearning.

We refer the reader to Appendix A for related work and a comparison with CPU (Shi et al., 2025), to Appendix I.2 for results on CIFAR100 and 20NewsGroups, targeted *class*-wise forgetting, MIA and unlearning time, and the sensitivity to c, d, α, and $|\mathcal{D}_f|$, to Appendix I.3 for the limitations of our framework, and to Appendix J for the sensitivity to the hyperparameters of EFACE together with the set-size ablation.

6 Conclusion

We have introduced a novel perspective on MU by anchoring it in the framework of CP, enabling a rigorous unlearning notion that is universal and unlinked to RT baselines. By defining conformal MU and corresponding empirical metrics, we offer a principled approach to evaluate unlearning effectiveness through the exclusion of forget data and the retention of coverage over retained data. The conformal approach extends to any task for which a nonconformity score can be defined, including regression, graph neural networks, and natural language models, as we discuss in Appendix B. Future work could explore tighter theoretical guarantees, adaptive methods tailored to diverse model architectures, and broader metrics to capture various dimensions of data influence.

7 AI use statement

AI tools have been used in polishing the writing of this manuscript by removing all typos and grammar mistakes as well as ensuring consistency and coherence. In addition, AI tools have been used to assist in writing the codes used in the experiments. Generative AI tools have NOT been used for any form of generating synthetic data or in proposing or refining hypotheses, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, designing or providing feedback on research methodology or experiments,

or in interpreting results. The assist with translation, cleaning and reformatting datasets, support in qualitative and thematic data analysis are not applicable to this work. The authors reviewed all AI-assisted work. All codes were checked, edited, and modified by the first author. Any polished text was edited by the first author to ensure readability and quality of presentation. The authors take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

8 Ethics Statement

The authors confirm that this research does not present any notable ethical concerns. Specifically, this study did not involve human participants or animal subjects. Furthermore, the work does not introduce or develop dual-use technologies, nor does it raise significant concerns regarding discrimination, bias, or algorithmic fairness.

9 Reproducibility statement

The authors have made extensive efforts to make this work reproducible. All assumptions used in theoretical results are provided. The proofs of all theoretical results are given in the appendix. For the experiment, all the details of the tests including the baselines and datasets with proper citations, models, hardware requirement, random seed specification, hyperparameters clarification for all baselines for all tables, and the basis of our implementation of baselines were all provided concisely in the main text and comprehensively in the appendices. The authors did not leave any details they believe is necessary for the reproducibility of the results but they included it in a dedicated section in the appendices (cf. Appendix I.1).

References

Youssef Allouah, Joshua Kazdan, Rachid Guerraoui, and Sanmi Koyejo. The utility and complexity of in- and out-of-distribution machine unlearning. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/pdf?id=HVFMooKrHX.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101. URL https://doi.org/10.1561/2200000101.

Yu Bai, Song Mei, Huan Wang, Yingbo Zhou, and Caiming Xiong. Efficient and differentiable conformal prediction with general function classes. In *International Conference on Learning Representations (ICLR)*, 2022. URL https://openreview.net/forum?id=Ht85_jyihxp. ICLR 2022.

Umit Yigit Basaran, Sk Miraj Ahmed, Amit Roy-Chowdhury, and Basak Guler. A certified unlearning approach without access to source data. In *International Conference on Machine Learning (ICML)*, 2025. URL https://openreview.net/pdf?id=8lt5776GLB.

Jeremy Bernstein, Arash Vahdat, Yisong Yue, and Ming-Yu Liu. On the distance between two neural networks and the stability of learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 21370–21381. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/f4b31bee138ff5f7b84ce1575a738f95-Paper.pdf.

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 (S&P)*, pp. 141–159, 2021. doi: 10.1109/SP40001.2021.00019.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *IEEE Symposium on Security and Privacy (S&P)*, pp. 463–480. IEEE, 2015. doi: 10.1109/SP.2015.35. URL https://ieeexplore.ieee.org/document/7163042.

Sungmin Cha, Sungjun Cho, Dasol Hwang, and Moontae Lee. Towards robust and parameter-efficient knowledge unlearning for LLMs. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/pdf?id=1ExfUpmIW4.

Min Chen, Weizhuo Gao, Gaoyang Liu, Kai Peng, and Chen Wang. Boundary unlearning: Rapid forgetting of deep networks via shifting the decision boundary. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7766–7775, 2023. doi: 10.1109/CVPR52729.2023.00750.

Victor Chernozhukov, Kaspar Wüthrich, and Yinchu Zhu. Exact and robust conformal inference methods for predictive machine learning with dependent data. In Sébastien Bubeck, Vianney Perchet, and Philippe Rigollet (eds.), *Conference on Learning Theory (COLT)*, volume 75 of *Proceedings of Machine Learning Research*, pp. 732–749. PMLR, Jul 2018. URL https://proceedings.mlr.press/v75/chernozhukov18a.html.

Eli Chien, Haoyu Peter Wang, Ziang Chen, and Pan Li. Certified machine unlearning via noisy stochastic gradient descent. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. URL https://openreview.net/pdf?id=h3k2NXu5bJ.

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)*. AAAI Press, 2023a. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i6.25879. URL https://doi.org/10.1609/aaai.v37i6.25879.

Vikram S. Chundawat, Ayush K. Tarun, Murari Mandal, and Mohan Kankanhalli. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 18:2345–2354, January 2023b. ISSN 1556-6013. doi: 10.1109/TIFS.2023.3265506. URL https://doi.org/10.1109/TIFS.2023.3265506.

Jase Clarkson. Distribution free prediction sets for node classification. In *Learning on Graphs Conference (LoG)*, Virtual Event, Dec. 9–12 2022. URL https://openreview.net/forum?id=Zg8y2-v8ia.

A. Feder Cooper, Christopher A. Choquette-Choo, Miranda Bogen, Kevin Klyman, Matthew Jagielski, Katja Filippova, Ken Liu, Alexandra Chouldechova, Jamie Hayes, Yangsibo Huang, Eleni Triantafillou, Peter Kairouz, Nicole Elyse Mitchell, Niloofar Mireshghallah, Abigail Z. Jacobs, James Grimmelmann, Vitaly Shmatikov, Christopher De Sa, Ilia Shumailov, Andreas Terzis, Solon Barocas, Jennifer Wortman Vaughan, danah boyd, Yejin Choi, Sanmi Koyejo, Fernando Delgado, Percy Liang, Daniel E. Ho, Pamela Samuelson, Miles Brundage, David Bau, Seth Neel, Hanna Wallach, Amy B. Cyphert, Mark Lemley, Nicolas Papernot, and Katherine Lee. Machine unlearning doesn't do what you think: Lessons for generative AI policy and research. In *Advances in Neural Information Processing Systems (NeurIPS), Position Paper Track*, 2025. URL https://openreview.net/pdf?id=mfd6GRW4Az.

Cynthia Dwork and Aaron Roth. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3–4):211–407, August 2014. ISSN 1551-305X. doi: 10.1561/0400000042. URL https://doi.org/10.1561/0400000042.

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 *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=gn0mIhQGNM.

Jack Foster, Stefan Schoepf, and Alexandra Brintrup. Fast machine unlearning without retraining through selective synaptic dampening. In *AAAI Conference on Artificial Intelligence (AAAI)*. Association for the Advancement of Artificial Intelligence, 2024. URL https://ojs.aaai.org/index.php/AAAI/article/view/29092/30065.

Jack Foster, Kyle Fogarty, Stefan Schoepf, Zack Dugue, Cengiz Oztireli, and Alexandra Brintrup. An information theoretic approach to machine unlearning. 2025. URL https://openreview.net/forum?id=t1utIThKHD.

Rina Foygel Barber, Emmanuel J. Candès, Aaditya Ramdas, and Ryan J. Tibshirani. Conformal prediction beyond exchangeability. *The Annals of Statistics*, 51(2):816–845, 2023. doi: 10.1214/23-AOS2276. URL https://projecteuclid.org/journals/annals-of-statistics/volume-51/issue-2/Conformal-prediction-beyond-exchangeability/10.1214/23-AOS2276.full.

Antonio A. 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)*, volume 32, 2019. URL https://openreview.net/pdf?id=H1xovVBg8B.

Laura Graves, Vineel Nagisetty, and Vijay Ganesh. Amnesiac machine learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 35, pp. 11516–11524. AAAI Press, 2021. doi: 10.1609/aaai.v35i13.17371. URL https://ojs.aaai.org/index.php/AAAI/article/view/17371.

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)*, volume 119, pp. 3832–3842. PMLR, 2020. URL https://proceedings.mlr.press/v119/guo20c.html.

Soroush H. Zargarbashi, Simone Antonelli, and Aleksandar Bojchevski. Conformal prediction sets for graph neural networks. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett (eds.), *International Conference on Machine Learning (ICML)*, volume 202 of *Proceedings of Machine Learning Research*, pp. 12292–12318. PMLR, 23–29 Jul 2023. URL https://proceedings.mlr.press/v202/h-zargarbashi23a.html.

Hsiang Hsu, Pradeep Niroula, Zichang He, Ivan Brugere, Freddy Lecue, and Chun-Fu Chen. The unseen threat: Residual knowledge in machine unlearning under perturbed samples. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/pdf?id=1wlZoi4w5o.

Kexin Huang, Ying Jin, Emmanuel J. Candès, and Jure Leskovec. Uncertainty quantification over graph with conformalized graph neural networks. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023. URL https://openreview.net/forum?id=ygjQCOyNfh. NeurIPS 2023 Spotlight.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning (ICML)*, volume 70 of *Proceedings of Machine Learning Research*, pp. 1885–1894. PMLR, 2017. URL https://proceedings.mlr.press/v70/koh17a.html.

Anastasia Koloskova, Youssef Allouah, Animesh Jha, Rachid Guerraoui, and Sanmi Koyejo. Certified unlearning for neural networks. In *International Conference on Machine Learning (ICML)*, 2025. URL https://openreview.net/forum?id=3rWQlV3s1I.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, 2009.

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

A. Lambrou, H. Papadopoulos, and A. Gammerman. Reliable confidence measures for medical diagnosis with evolutionary algorithms. *IEEE Transactions on Information Technology in Biomedicine*, 15(1):93–99, January 2011. ISSN 1089-7771. doi: 10.1109/TITB.2010.2091144. URL https://doi.org/10.1109/TITB.2010.2091144.

Ken Lang. Newsweeder: Learning to filter netnews. International Conference on Machine Learning (ICML), 1995.

Jing Lei, Max G'Sell, Alessandro Rinaldo, Ryan J. Tibshirani, and Larry Wasserman. Distribution-free predictive inference for regression. *Journal of the American Statistical Association*, 113 (523):1094–1111, 2018. doi: 10.1080/01621459.2017.1307116. URL https://doi.org/10.1080/01621459.2017.1307116.

Sijia Liu, Yuanshun Yao, Jinghan Jia, Stephen Casper, Nathalie Baracaldo, Peter Hase, Yuguang Yao, Chris Yuhao Liu, Xiaojun Xu, Hang Li, Kush R. Varshney, Mohit Bansal, Sanmi Koyejo, and Yang Liu. Rethinking machine unlearning for large language models. *Nature Machine Intelligence*, 7: 181–194, 2025. doi: 10.1038/s42256-025-00985-0.

Charles Lu, Andréanne Lemay, Ken Chang, Katharina Höbel, and Jayashree Kalpathy-Cramer. Fair conformal predictors for applications in medical imaging. *AAAI Conference on Artificial Intelligence (AAAI)*, 36(11):12008–12016, Jun. 2022. doi: 10.1609/aaai.v36i11.21459. URL https://ojs.aaai.org/index.php/AAAI/article/view/21459.

Siqiao Mu and Diego Klabjan. Rewind-to-delete: Certified machine unlearning for nonconvex functions. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. URL https://openreview.net/pdf?id=FgjcLXIUjr.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *International Conference on Algorithmic Learning Theory (ALT)*, volume 132 of *Proceedings of Machine Learning Research*, pp. 931–962. PMLR, 2021. URL https://proceedings.mlr.press/v132/neel21a.html.

Quoc Phong Nguyen, Bryan Kian Hsiang Low, and Patrick Jaillet. Variational bayesian unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pp. 20865–20876. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/b8a6550662b363eb34145965d64d0cfb-Paper.pdf.

Zhuoyi Peng, Yixuan Tang, and Yi Yang. Adversarial mixup unlearning. In *International Conference on Learning Representations (ICLR)*, 2025. URL https://openreview.net/forum?id=GcbhbZsgiu.

Aleksandr Podkopaev and Aaditya Ramdas. Distribution-free uncertainty quantification for classification under label shift. In Cassio de Campos and Marloes H. Maathuis (eds.), *Conference on Uncertainty in Artificial Intelligence (UAI)*, volume 161 of *Proceedings of Machine Learning Research*, pp. 844–853. PMLR, Jul 2021. URL https://proceedings.mlr.press/v161/podkopaev21a.html.

Victor Quach, Adam Fisch, Tal Schuster, Adam Yala, Jae Ho Sohn, Tommi S. Jaakkola, and Regina Barzilay. Conformal language modeling. In *International Conference on Learning Representations (ICLR)*, 2024. URL https://openreview.net/forum?id=pzUhfQ74c5.

Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525):223–234, Jan. 2019. doi: 10.1080/01621459.2017.1395341. URL https://doi.org/10.1080/01621459.2017.1395341.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*. Curran Associates, Inc., 2021. URL https://openreview.net/pdf?id=pvCLqcsLJ1N.

Glenn Shafer and Vladimir Vovk. *Conformal Prediction for Reliable Machine Learning: Theory, Adaptations, and Applications*. Elsevier, 2008. ISBN 9780123985378.

Ambesh Shekhar. Imagenet100: A sample of imagenet classes. Kaggle Dataset, August 2021. URL https://www.kaggle.com/datasets/ambityga/imagenet100.

Yingdan Shi, Sijia Liu, and Ren Wang. Rethinking evaluation metrics for machine unlearning. In *ICML 2025 Workshop on Machine Unlearning for Generative AI*, 2025. URL https://openreview.net/forum?id=wuGgok1Zyd.

Ayush K. Tarun, Vikram S. Chundawat, Murari Mandal, and Mohan S. Kankanhalli. Fast Yet Effective Machine Unlearning. *IEEE Transactions on Neural Networks and Learning Systems*, 35(9):13046–13055, September 2024. doi: 10.1109/TNNLS.2023.3266233. URL https://ieeexplore.ieee.org/document/10113700/.

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)*, pp. 303–319, 2022a. doi: 10.1109/EuroSP53844.2022.00027.

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)*, pp. 4007–4022, Boston, MA, 2022b. USENIX Association. URL https://www.usenix.org/conference/usenixsecurity22/presentation/thudi.

Ryan J. Tibshirani, Rina Foygel Barber, Emmanuel J. Candès, and Aaditya Ramdas. Conformal prediction under covariate shift. In *Advances in Neural Information Processing Systems (NeurIPS)*, Red Hook, NY, USA, 2019. Curran Associates Inc.

Eleni Triantafillou, Ahmed Imtiaz Humayun, Monica Ribero, Alexander Matt Turner, Michael C. Mozer, and Georgios Kaissis. Is your algorithm unlearning or untraining?, 2026. URL https://arxiv.org/abs/2604.07962.

Daniel Trippa, Cesare Campagnano, Maria Sofia Bucarelli, Gabriele Tolomei, and Fabrizio Silvestri. Gradient-based and task-agnostic machine unlearning. In *International Conference on Machine Learning (ICML)*. PMLR, 2024. URL https://openreview.net/pdf?id=31xHtxKmOP.

Vladimir Vovk, Alex Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, New York, 2005.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. Machine unlearning of features and labels. In *Network and Distributed System Security Symposium (NDSS)*. Internet Society, 2023. URL https://www.ndss-symposium.org/wp-content/uploads/2023/02/ndss2023_s87_paper.pdf.

Shizhou Xu and Thomas Strohmer. Machine unlearning via information theoretic regularization. *arXiv preprint arXiv:2502.05684*, 2025.

Dawen Zhang, Pamela Finckenberg-Broman, Thong Hoang, Shidong Pan, Zhenchang Xing, Mark Staples, and Xiwei Xu. Right to be forgotten in the era of large language models: Implications, challenges, and solutions. *AI and Ethics*, September 2024. doi: 10.1007/s43681-024-00573-9. URL https://link.springer.com/article/10.1007/s43681-024-00573-9.

A Related Work

Existing machine unlearning methods face significant challenges when applied to CP, including (I1) reliance on RT model baselines for evaluation and (I2) vulnerability to forgeability in parameter-space definitions. Below, we review relevant literature and highlight these limitations.

Most unlearning approaches aim to approximate a model retrained from scratch without the forget data (Cao & Yang, 2015; Guo et al., 2020; Ginart et al., 2019; Chien et al., 2024; Koloskova et al., 2025). These methods often assume strong convexity (Sekhari et al., 2021; Allouah et al., 2025; Thudi et al., 2022a) or employ practical mechanisms such as gradient-influence subtraction (Warnecke et al., 2023; Graves et al., 2021), Bayesian updates (Nguyen et al., 2020), teacher–student transfer (Chundawat et al., 2023a), noise-based unlearning (Chundawat et al., 2023b), and information-theoretic objectives (Foster et al., 2025; Xu & Strohmer, 2025). Other strategies include KL-divergence–based forgetting (Kurmanji et al., 2023), selective gradient dampening (Trippa et al., 2024; Foster et al., 2024), saliency-guided relabeling (Fan et al., 2024), decision-boundary shifting (Chen et al., 2023), and adversarial mixup (Peng et al., 2025). Recent work addresses "residual knowledge" by penalizing prediction deviations on neighboring samples of the forget data (Hsu et al., 2025), but the objective remains to approximate a RT model.

Certified unlearning methods, such as (Koloskova et al., 2025), relax this goal by approximating any certifying model trained without the forget data. However, these approaches are susceptible to forgeability (Thudi et al., 2022b), as even the original model's parameters can satisfy such definitions.

A common approach in empirical evaluation of unlearning is to test the unlearning of random instances. By letting $W = (X, Y)$ and setting $\mathcal{W}_{\text{forget}}$ to correspond to a specific subset of training data (cf. Section 2), we recover the standard specific instance forgetting setup (Cao & Yang, 2015; Ginart et al., 2019; Warnecke et al., 2023; Sekhari et al., 2021; Guo et al., 2020). However, specific instance forgetting does not align with our intended focus on conformal unlearning, which leverages shared characteristics to define the forget set. In the case of isolated instances, unlearning is vacuous since the model is generalizable, leading to fake conformal unlearning (Shi et al., 2025). In addition, to evaluate the performance of specific instance forgetting, random forgetting (Foster et al., 2024; 2025; Shi et al., 2025; Peng et al., 2025; Graves et al., 2021) is often employed, where $\mathcal{D}_f$ is a random sub-sample of $\mathcal{D}_{\text{train}}$, which violates the spirit of conformal unlearning. Therefore, in this work, we do not perform any random forgetting experiments.

The first conformal-prediction–based evaluation for unlearning was proposed by (Shi et al., 2025), which introduced the metric

$$\mathsf{CR}(\mathcal{D}) \triangleq \frac{\sum_{(x,y)\in\mathcal{D}} \mathbf{1}\{y \in \mathcal{C}(X)\}}{\sum_{(x,y)\in\mathcal{D}} |\mathcal{C}(X)|}, \tag{16}$$

targeting low CR on $\mathcal{D}_f$ and high CR on $\mathcal{D}_r$. However, CR has notable limitations: (i) it may underestimate coverage on forget data due to large denominators, (ii) it can overestimate coverage on retained data with small prediction sets, and (iii) it is less interpretable in many-class settings due to uniformly small values. Empirically, CR sometimes fails to distinguish between forget and retain sets (Shi et al., 2025). In contrast, our proposed metrics, C and M, directly measure coverage and miscoverage rates, aligning with the theoretical framework of (α, β)-conformal unlearning (Definition 1).

Table 4: ImageNet100, ResNet18 10 classes forgetting with $c, d = 100$ and $\alpha = 0.05$.

Metric	CPU	EFACE
$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	1.00 ± 0.00	1.00 ± 0.00
$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	0.02 ± 0.00	0.17 ± 0.01
$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	0.96 ± 0.00	0.98 ± 0.00
$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	0.07 ± 0.01	0.27 ± 0.02
$\mathsf{CR}(\mathcal{V}_r)$	0.04 ± 0.00	0.03 ± 0.00
$\mathsf{CR}(\mathcal{V}_f)$	0.02 ± 0.00	0.01 ± 0.00
$H \uparrow$	0.06 ± 0.01	0.34 ± 0.02

Table 4 compares EFACE with the CPU (fine-tuning) variant (Shi et al., 2025) and the CR metric (cf. (16)). EFACE consistently meets the desired C/M targets, achieving a significant H improvement (≥ 0.28) over CPU. In contrast, the CR metric can be misleading: smaller prediction sets reduce the denominator, potentially inflating $\mathsf{CR}(\mathcal{V}_f)$ even when true labels are frequently covered (e.g., EFACE's $\mathsf{M}(\mathcal{V}_f) = 0.27 > 0.07 =$ CPU's $\mathsf{M}(\mathcal{V}_f)$). Similarly, $\mathsf{CR}(\mathcal{V}_r)$ may appear disproportionately small in many-class settings despite high retained coverage (e.g., $\mathsf{C}(\mathcal{V}_r) \approx 0.98$). These results verify that *CR does not reliably address fake conformal unlearning*, particularly in scenarios with large label spaces.

Methodologically, the CPU procedure in (Shi et al., 2025) updates conformity scores only for forget data and finds the quantile score using held-out set during unlearning, whereas EFACE optimizes the miscoverage gap between forget and retain sets, and only uses the $\mathcal{D}_{\text{unlearn}}$ for unlearning. By grounding unlearning in explicit coverage and miscoverage targets, EFACE addresses **I1**–**I3**.

Finally, $\nabla\tau$ (Trippa et al., 2024) uses external data to induce forgetting by matching entropy losses between validation and training forget sets. However, it does not leverage conformity scores or target conformal objectives, leading to performance differences highlighted in our results.

By defining forget data through shared characteristics, conformal unlearning also mitigates the *residual knowledge* problem identified in prior unlearning methods (Hsu et al., 2025). Residual knowledge arises when an unlearned model's predictions align with those of a RT model on the exact forget data points but deviate on slightly perturbed samples in their neighborhood, indicating that latent information about the forget data persists. In conformal unlearning, the shared characteristics

defining the forget data ensure that small perturbations are likely to remain within the forget set, naturally extending the unlearning process to the local neighborhood of the forget data.

This intuition can be formalized under mild regularity conditions. Suppose the scoring function s is M-Lipschitz continuous with respect to its first argument. For any $X \sim p_{\text{data}}$ and a perturbed input $X' = X + \delta$, where $\|\delta\| \leq \epsilon$ for some small $\epsilon > 0$, it follows that

$$|s(X, Y) - s(X', Y)| \leq M\|X - X'\| \leq M\epsilon \text{ a. s.}$$

By Proposition E.1, if the cumulative distribution function (cdf) of s is L-Lipschitz, the perturbed scores $s(X', Y)$ satisfy (5) with $\alpha' \leq \alpha + \frac{2LM\epsilon}{\pi_r}$ and (6) with $\beta' \geq \beta - \frac{2LM\epsilon}{\pi_f}$. Therefore, for scoring functions that exhibit sufficient smoothness and for small perturbations, the guarantees of conformal unlearning degrade in a controlled manner within the neighborhood of the forget data. This directly mitigates the issue of residual knowledge by ensuring that the unlearning guarantees extend to local perturbations of the forget data.

B How Can Conformal Machine Unlearning Be Generalized?

We have focused on machine unlearning for regression and classification tasks. However, our work establishes a foundational framework for a new paradigm of machine unlearning based on rigorous quantification of conformal prediction uncertainty. This paradigm naturally extends to other tasks through appropriate non-conformity score functions, as demonstrated in foundational conformal prediction literature (cf. (Angelopoulos & Bates, 2023; Shafer & Vovk, 2008)). Furthermore, the approach generalizes to any domain where conformal prediction has been developed with suitable handling of its theoretical foundations. For instance, conformal prediction has been successfully applied to graph neural networks (H. Zargarbashi et al., 2023; Huang et al., 2023; Clarkson, 2022), natural language processing (Cha et al., 2025; Quach et al., 2024; Liu et al., 2025; Zhang et al., 2024), and other emerging application areas. Consequently, our paradigm is immediately applicable to contemporary machine unlearning research across these domains. Extending the framework to these additional areas represents a promising direction for future work.

C The Regression Case

Consider a regression model where the data $(X, Y) \in \mathcal{X} \times \mathbb{R}^m$ and $\mathcal{X}$ is a subset of a Euclidean space. The relationship between the input X and the output Y is modeled as $Y = f_\theta(X) + \varepsilon$, where $f_\theta : \mathcal{X} \to \mathbb{R}^m$ denotes the regression function parameterized by $\theta \in \mathbb{R}^d$, and ε represents the noise term. Here, d specifies the dimensionality of the parameter space. This is an example of feature-based cluster-wise forgetting. We assume that $\mathcal{X} = \mathcal{X}_R \cup \mathcal{X}_F$, where $\mathcal{X}_R$ and $\mathcal{X}_F$ are *disjoint* feature subspaces corresponding to the retain and forget groups, respectively.[1] We show that under some technical conditions, we can achieve perfect conformal unlearning (i.e., $\beta = 1$) in this example. Suppose that θ_o is the model parameter learned from training data before the unlearning procedure.

Following (Vovk et al., 2005; Angelopoulos & Bates, 2023), we choose the score function as

$$s(X, Y; \theta) \triangleq \|Y - f_\theta(X)\|.$$

Let n be the size of the calibration set $\mathcal{D}_{\text{calib}}$. The conformal predictor associated with θ is

$$\mathcal{C}_\theta(X) \triangleq \{y \in \mathbb{R}^m : \|y - f_\theta(X)\| \leq \hat{q}_\alpha(\theta)\}, \tag{17}$$

where $\hat{q}_\alpha(\theta)$ is the $\lceil (1-\alpha)(n+1) \rceil$ quantile of the scores computed from $\mathcal{D}_{\text{calib}}$. From (Lei et al., 2018), we have

$$1 - \alpha \leq \mathbb{P}(Y \in \mathcal{C}_\theta(X)) \leq 1 - \alpha + \frac{1}{n+1}. \tag{18}$$

The following assumptions are imposed to facilitate the analysis. For clarity, we use $\nabla_\theta f_{\theta_o} \triangleq \nabla_\theta f_\theta \mid_{\theta=\theta_o}$ to denote the Jacobian of f_θ with respect to θ, evaluated at $\theta = \theta_o$.

[1] For a slightly more complex model, we can impose this assumption on the semantic space obtained after projecting raw features through a deep neural network. The analysis remains similar in that case.

Assumption C.1. *(i)* Continuous variables. (X, Y) *are continuous random variables, i.e., their joint distribution admits a probability density function.*

(ii) Bounded noise. *We have* $\|\varepsilon\| \leq \sigma$ *almost surely (*a. s.*) for some* $\sigma > 0$.

(iii) Smoothness with respect to θ. *The gradient of the parameterized model* f_θ *is assumed to be* L*-Lipschitz continuous with respect to* θ *in a neighborhood around* θ_o*. Formally, for all* $x \in \mathcal{X}$*, the following condition holds:*

$$\|\nabla_\theta f_\theta(x) - \nabla_\theta f_{\theta_o}(x)\| \leq L\|\theta - \theta_o\|,$$

where $L > 0$ *is the Lipschitz constant. This assumption implies that the model can be locally approximated by its first-order Taylor expansion around* θ_o *with bounded error.*

(iv) Directional separability in parameter space. *There exists a unit vector* $v \in \mathbb{R}^d$ *(i.e.,* $\|v\| = 1$*) such that, for some constants* $0 \leq a_R \leq a_F$*, the following conditions hold:*

$$\sup_{x \in \mathcal{X}_R} \|\nabla_\theta f_{\theta_o}(x)v\| \leq a_R,$$
$$\inf_{x \in \mathcal{X}_F} \|\nabla_\theta f_{\theta_o}(x)v\| \geq a_F.$$

Denote $\Delta a = a_F - a_R > 0$*. Assume that* $(\Delta a)^2 \geq 8\sigma L$*. In other words, the model* f_{θ_o} *exhibits limited variation in the direction* v *within the retain feature space* $\mathcal{X}_R$*, while demonstrating significant variation in the same direction within the forget feature space* $\mathcal{X}_F$*. Furthermore, the difference in variation between the two feature spaces is sufficiently large relative to the noise level* σ *and the Lipschitz constant* L*.*

(v) Retained subpopulation mass. *The proportion of the retained subpopulation satisfies*

$$\mathbb{P}(\mathcal{X}_R) > 1 - \alpha + \frac{1}{n+1}.$$

Theorem C.1. *[Achievability of (*α*,1)-conformal unlearning in regression.] Suppose Assumption C.1 holds with* θ_o *being the global minimizer of the population loss before unlearning. Then, there exists a step size* $\gamma^* \in (\gamma_-, \gamma_+)$ *with*

$$\gamma_\pm = \frac{\Delta a \pm \sqrt{(\Delta a)^2 - 8\sigma L}}{2L}, \tag{19}$$

such that the unlearned parameters $\theta_u = \theta_o + \gamma^* v$ *achieve (*α*,*β*)-conformal unlearning with* $\beta = 1$ *for the CP procedure defined by* (17)*.*

Proof. Let $\theta_\gamma = \theta_o + \gamma v$, where v is the unit vector from Assumption C.1(iv). Then by Assumption C.1(iii) and the Taylor expansion around θ_o, we have a. s.,

$$\begin{aligned} Y - f_{\theta_o+\gamma v}(X) &= f_{\theta_o}(X) + \varepsilon - f_{\theta_o+\gamma v}(X) \\ &= \varepsilon - \gamma \nabla_\theta f_{\theta_o}(X)v - r(X, \gamma). \end{aligned}$$

where the remainder $\|r(X, \gamma)\| \leq \frac{1}{2}L\gamma^2$.

Using the triangle inequality and Assumption C.1(ii), for any $X \in \mathcal{X}_R$, we have

$$\begin{aligned} \|Y - f_{\theta_o+\gamma v}(X)\| &\leq |\gamma| \|\nabla_\theta f_{\theta_o}(X)v\| + \sigma + \frac{1}{2}L\gamma^2 \\ &\leq |\gamma| a_R + \sigma + \frac{1}{2}L\gamma^2 \triangleq r_R^{\max}(\gamma). \end{aligned} \tag{20}$$

Similarly, for any $X \in \mathcal{X}_F$, we have

$$\|Y - f_{\theta_o+\gamma v}(X)\| \geq |\gamma| a_F - \sigma - \frac{1}{2}L\gamma^2 \triangleq r_F^{\min}(\gamma). \tag{21}$$

If there exists a step size γ^* such that $r_R^{\max}(\gamma^*) < r_F^{\min}(\gamma^*)$, then we have

$$\begin{aligned} &\mathbb{P}(s(X, Y; \theta_{\gamma^*}) \leq r_R^{\max}(\gamma^*)) = \mathbb{P}(\{(X, Y) : X \in \mathcal{X}_R\}) \\ &\geq 1 - \alpha + \frac{1}{n+1} \geq \mathbb{P}(s(X, Y; \theta_{\gamma^*}) \leq \hat{q}_\alpha(\theta_{\gamma^*})), \end{aligned}$$

where the first inequality follows from Assumption C.1(v) and the second inequality from (18). Hence, $\hat{q}_\alpha(\theta_{\gamma^*}) \le r_R^{\max}(\gamma^*) < r_F^{\min}(\gamma^*)$ a. s. Therefore, $\mathbb{P}\big(Y \in \mathcal{C}_{\theta_{\gamma^*}}(X) \,\big|\, X \in \mathcal{X}_F\big) = 0$, or $\beta = 1$, while the $1-\alpha$ coverage is retained on $X \in \mathcal{X}_R$ by Lemma 1. Thus, $\theta_u = \theta_{\gamma^*}$ achieves (α,1)-conformal unlearning.

To find such a step size γ^*, letting $r_R^{\max}(\gamma) < r_F^{\min}(\gamma)$, we obtain

$$|\gamma|(a_F - a_R) - L\gamma^2 > 2\sigma \tag{22}$$

$$L\gamma^2 - |\gamma|\Delta a + 2\sigma < 0. \tag{23}$$

Taking $\gamma \ge 0$, the inequality in (23) is satisfied for $\gamma \in (\gamma_-, \gamma_+)$, where $\gamma_\pm$ are defined in (19). The proof is now complete by choosing $\gamma^* \in (\gamma_-, \gamma_+)$. □

Numerical illustration To confirm that the perfect-forgetting regime promised by Theorem C.1 is realizable, we instantiate the regression setting based on Assumption C.1. We generate data (X, Y) with $Y = f_{\theta_o}(X) + \varepsilon$ from a model whose parameter gradient is L-Lipschitz (Assumption C.1(iii)), and split the feature space into a retain cluster $\mathcal{X}_R$ and a forget cluster $\mathcal{X}_F$ whose directional gradients along v at θ_o satisfy $\|\nabla_\theta f_{\theta_o}(x)v\| = a_R = 0.5$ on $\mathcal{X}_R$ and $a_F = 2.0$ on $\mathcal{X}_F$ (Assumption C.1(iv)). The additive noise is bounded, $\varepsilon \sim \mathrm{Unif}[-\sigma, \sigma]$ with $\sigma = 0.1$ (Assumption C.1(ii)), the Lipschitz constant is $L = 1$, and the retained subpopulation mass is $\mathbb{P}(\mathcal{X}_R) = 0.95 \ge 1 - \alpha + \frac{1}{n+1}$ (Assumption C.1(v)) with $\alpha = 0.1$. These constants satisfy the margin condition $(\Delta a)^2 = 2.25 \ge 8\sigma L = 0.8$, so (19) yields the interval $(\gamma_-, \gamma_+) = (0.148, 1.352)$. Using $n = 2000$ calibration points to form the split-conformal threshold $\hat{q}_\alpha$ and 200,000 independent test set to estimate coverage, we set $\theta_u = \theta_o + \gamma v$ and sweep γ. As shown in Fig. 3, the forget-set miscoverage attains $\beta = 1$ precisely on (γ_-, γ_+), every forgotten point is excluded from its prediction set $\mathcal{C}_{\theta_u}(X)$ of (17), while the retain-set coverage never drops below $1 - \alpha$. Averaged over 10 random draws, at the interval midpoint $\gamma^* = 0.75$ we measure $\beta = 1.00 \pm 0.00$ and retain coverage 0.947 ± 0.006. This confirms that the perfect-forgetting step guaranteed by Theorem C.1 is realizable in practice, under the given assumptions.

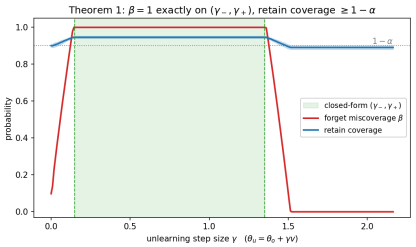


Figure 3: Numerical illustration of Theorem C.1. Starting from the pre-unlearning minimizer θ_o, we take the prescribed step $\theta_u = \theta_o + \gamma v$ and sweep the step size γ, reporting the forget-set miscoverage β (red) and the retain-set coverage (blue) of the split-CP procedure in (17) (mean over 10 random draws; shaded region ± 1 standard deviation). The green band is the closed-form interval (γ_-, γ_+) of (19). Perfect forgetting $\beta = 1$ holds *exactly* over the interval, while the retain coverage remains above $1 - \alpha$.

D EFACE ALGORITHM

This appendix gives the practical relaxations of (15) that EFACE solves, followed by its pseudo-code. To address the non-differentiability of the indicator function, we employ a sigmoid hinge approximation, which facilitates gradient-based optimization. In that case, $\widehat{L}_{\mathrm{cov},\mathcal{D}}(\mathcal{C}_{\theta,t}) \approx \widehat{L}'_{\mathrm{cov},\mathcal{D}}(\mathcal{C}_{\theta,t}) = \frac{1}{|\mathcal{D}|}\sum_{(x,y)\in\mathcal{D}} \sigma\left(\kappa \cdot (s(x,y) - t)\right)$. Moreover, in practice, the model might be prone to learning an easy threshold t to minimize the objective in (15). We found that choosing $\hat{t} = \hat{q}_\alpha$ of the conformity scores of $\mathcal{D}_{\mathrm{unlearn}}$ at each epoch during unlearning yields better generalization empirically but requires the level α as an input to the algorithm. Additionally, a regularization term, $\gamma\|\theta_o - \theta_u\|^2$, is incorporated into the objective function to mitigate excessive deviation of the unlearned model's parameters from the original model's parameters, thereby preserving utility. The steepness parameter of the sigmoid hinge, κ, and the regularization coefficient, γ, are treated as hyperparameters and are subject to fine-tuning. Following these relaxations, the aim is to solve the following updated minimization problem.

$$\min_\theta \widehat{L}'_{\mathrm{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,\hat{t}}) - \widehat{L}'_{\mathrm{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,\hat{t}}) + \gamma\|\theta_o - \theta\|^2. \tag{24a}$$

The impact of these hyperparameters, is analyzed in the sensitivity analysis provided in Appendix J. Our EFACE algorithm is summarized in the pseudo-code in Algorithm 1. Note that $\mathcal{D}_{\mathrm{calib}}$ is used for

Algorithm 1 EFACE

Require: Retained data $\mathcal{D}_r$, forget data $\mathcal{D}_f$, calibration data $\mathcal{D}_{\text{calib}}$ with $|\mathcal{D}_{\text{calib}}| = n$, conformity scoring function $s(X, Y)$, steepness κ, regularization constant γ, miscoverage tolerance α

1: Define $\mathcal{C}_{\theta,\hat{t}}(x) = \{y : s(x, y) \leq \hat{t}\}$
2: Solve (24) to obtain θ_u.
3: Compute $\hat{q}_\alpha$ as the $\lceil (1-\alpha)(n+1) \rceil$ quantile of $\{s(X, Y; \theta_u) : (X, Y) \in \mathcal{D}_{\text{calib}}\}$.
4: Set $\mathcal{C}_{\theta_u}(X) = \{y : s(X, y; \theta_u) \leq \hat{q}_\alpha\}$

Ensure: $\theta_u, \mathcal{C}_{\theta_u}(X)$

reconformalization only after unlearning is completely done, hence the exchangeability with $\mathcal{D}_f \cup \mathcal{D}_r$ is not violated.

D.1 Generalization Bounds

Let $(\widehat{\theta}, \widehat{t})$ denote a solution obtained from the ERM problem in (15). To quantify the generalization gap between the empirical and population-level metrics, we define the following conditional concentration terms:

$$\begin{aligned} \varepsilon_r &\triangleq \sup_{\theta,t} \Big| L_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) - \widehat{L}_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) \Big|, \\ \varepsilon_f &\triangleq \sup_{\theta,t} \Big| L_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}) - \widehat{L}_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}) \Big|. \end{aligned} \tag{25}$$

These terms characterize the maximum deviation between the empirical and true values of the miscoverage loss, conditioned on the retain and forget sets, respectively.

Proposition D.1 (Near-optimal conditional miscoverage *difference*)**.** *A solution* $(\widehat{\theta}, \widehat{t})$ *of* (15) *satisfies*

$$\begin{aligned} &L_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) - L_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) \\ &\leq \inf_{(\theta,t)} \Big(L_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) - L_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}) \Big) + 2\varepsilon_r + 2\varepsilon_f. \end{aligned} \tag{26}$$

Proof. We have

$$\begin{aligned} &L_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) - L_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) \\ &\leq \widehat{L}_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) - \widehat{L}_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\widehat{\theta},\widehat{t}}) + \varepsilon_r + \varepsilon_f \\ &\leq \widehat{L}_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) - \widehat{L}_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}) + \varepsilon_r + \varepsilon_f \\ &\leq L_{\text{cov},\mathcal{D}_r}(\mathcal{C}_{\theta,t}) - L_{\text{cov},\mathcal{D}_f}(\mathcal{C}_{\theta,t}) + 2\varepsilon_r + 2\varepsilon_f \end{aligned}$$

for any (θ, t). The first and last inequalities follow from the definitions of ε_r and ε_f, and the second inequality follows from the optimality of $(\widehat{\theta}, \widehat{t})$ for (15). Taking the infimum over (θ, t) gives the claim. □

Concrete bounds for the terms in (25) under finite/VC/Rademacher classes are provided in App. C of (Bai et al., 2022), leading to rates of order $\sqrt{\text{Comp}(\mathcal{C})/n_{\text{unlearn}}}$, where $\text{Comp}(\mathcal{C})$ is the complexity measure of the class $\mathcal{C}$. Therefore, by having a sufficiently large n_{unlearn}, these terms can be made arbitrarily small. We refer the reader to (Bai et al., 2022) for more details.

Remark D.1. *Proposition D.1 concerns a solution* $(\widehat{\theta}, \widehat{t})$ *to the empirical problem in* (15)*, assuming such a solution is obtained. It bounds the discrepancy between sample quantities (observed frequencies) and population quantities (the probabilities in* (13)*) via the uniform deviation terms in* (25)*. The result is therefore independent of both the optimizer and the architecture, and it does not assert that stochastic gradient descent on a deep network attains the minimizer of* (15)*. The explicit rates of order* $\sqrt{\text{Comp}(\mathcal{C})/n_{\text{unlearn}}}$ *for the terms in* (25)*, under finite, VC, or Rademacher classes, apply once an appropriate complexity measure is specified for the class under study.*

This secondary result is distinct from the paper's finite-sample claim. The finite-sample, uncertainty-aware guarantee used in the paper is the conformal guarantee in (4)*, together with Lemma 1 and*

Lemma 2. This guarantee is distribution-free, holds for any θ_u and any model class, and requires only the exchangeability condition in Assumption 1, which the protocol described in Appendix I.1 satisfies on the held-out evaluation data. It does not rely on Proposition D.1. Rather, Proposition D.1 addresses the separate question of how far a solution to the empirical problem may be from the population optimum.

E Different Conformity Score Function During Inference

EFACE employs a CP procedure $\mathfrak{CP}(\theta, s)$ to facilitate unlearning. However, a downstream practitioner may opt to utilize an alternative nonconformity score function s'. This raises a pertinent question: does achieving (α, β)-conformal unlearning with respect to s provide any guarantees when evaluated under s'?

Throughout this section, we assume that there exists an unlearning algorithm that achieves (α, β)-conformal unlearning with respect to s. We provide sufficient conditions under which the coverage and miscoverage guarantees exhibit controlled degradation when using s'. Specifically, if $\mathfrak{U}$ satisfies (α, β)-conformal unlearning with respect to s, then it also satisfies (α', β')-conformal unlearning with respect to s', where α' and β' remain close to α and β, respectively. We provide explicit bounds to quantify the extent of this degradation. We denote the CP prediction set constructed using s' as $\mathcal{C}'_{\theta_u}(\cdot)$ and the $\lceil (1-\alpha)(n+1) \rceil / n$ quantile of the s' scores on $\mathcal{D}_{\text{calib}}$ as $\hat{q}'_\alpha$. To avoid clutter, we write s and s' instead of $s(X, Y)$ and $s'(X, Y)$ for $(X, Y) \sim p_{\text{data}}$. For a score function s, let $F_s(\cdot) = \mathbb{P}(s(X, Y) \leq \cdot)$ denote its cdf, $F_{r,s}(\cdot) = \mathbb{P}_r(s(X, Y) \leq \cdot)$, and $F_{f,s}(\cdot) = \mathbb{P}_f(s(X, Y) \leq \cdot)$.

Lemma E.1. *Suppose $s' = g(s)$, for a strictly increasing function $g(\cdot)$. Then, $\alpha' = \alpha$, and $\beta' = \beta$. I.e., conformal unlearning guarantees are invariant under strictly increasing transformations of the conformity score.*

Proof. Using the score function s', the coverage probability on $\mathcal{V}_r$ is

$$\begin{aligned}
\mathbb{P}_r\big(Y \in \mathcal{C}'_{\theta_u}(X)\big) &= \mathbb{P}_r(s'(X, Y) \leq \hat{q}'_\alpha) \\
&= \mathbb{P}_r(g(s(X, Y)) \leq \hat{q}'_\alpha) \\
&= \mathbb{P}_r(g(s(X, Y)) \leq g(\hat{q}_\alpha)) \\
&= \mathbb{P}_r(s(X, Y) \leq \hat{q}_\alpha) \\
&= \mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) \geq 1 - \alpha,
\end{aligned}$$

where the third and fourth equalities follow since $g(\cdot)$ is strictly increasing. A similar proof yields

$$\mathbb{P}_f\big(Y \notin \mathcal{C}'_{\theta_u}(X)\big) \geq \beta.$$

□

The sufficient condition in Lemma E.1 is distribution-free but limited to strictly monotonic transformations. To handle more general perturbations, we impose distributional assumptions. The following propositions quantify how bounded deviations between s and s' affect the coverage guarantees.

Proposition E.1. *Suppose $|s' - g(s)| \leq c$ a. s., where $g(\cdot)$ is a strictly increasing function and $c \geq 0$ is a constant. In addition, suppose that $F_{s'}(\cdot)$ is L-Lipschitz. Then,*

$$\alpha' \leq \alpha + \frac{2Lc}{\pi_r}, \tag{27}$$

$$\beta' \geq \beta - \frac{2Lc}{\pi_f}. \tag{28}$$

Proof. Define a new score function $s'' = g(s)$. By Lemma E.1, the unlearning guarantees for s'' remain valid with the same α and β. Given the point-wise bounded difference between s'' and s', we have $F_{s''}(\sigma - c) \leq F_{s'}(\sigma) \leq F_{s''}(\sigma + c)$ for all σ. Moreover, it follows that $\hat{q}'_\alpha - c \leq \hat{q}''_\alpha \leq \hat{q}'_\alpha + c$ a. s..

With the expectation taken over $\mathcal{D}_{\text{calib}}$, we have

$$\begin{aligned}
&\mathbb{E}F_{r,s'}(\hat{q}'_\alpha + 2c) - \mathbb{E}F_{r,s'}(\hat{q}'_\alpha) \\
&= \mathbb{P}_r(\hat{q}'_\alpha < s' \le \hat{q}'_\alpha + 2c) \\
&\le \frac{\mathbb{P}(\hat{q}'_\alpha < s' \le \hat{q}'_\alpha + 2c)}{\pi_r} \\
&= \frac{\mathbb{E}[F_{s'}(\hat{q}'_\alpha + 2c) - F_{s'}(\hat{q}'_\alpha)]}{\pi_r} \\
&\le \frac{2Lc}{\pi_r},
\end{aligned}$$

where the last inequality follows from the L-Lipschitz assumption on $F_{s'}$. Therefore,

$$\begin{aligned}
\mathbb{E}F_{r,s'}(\hat{q}'_\alpha) &\ge \mathbb{E}F_{r,s'}(\hat{q}'_\alpha + 2c) - \frac{2Lc}{\pi_r} \\
&= \mathbb{P}_r(s' \le \hat{q}'_\alpha + 2c) - \frac{2Lc}{\pi_r} \\
&\ge \mathbb{P}_r(s'' \le \hat{q}'_\alpha + c) - \frac{2Lc}{\pi_r} \\
&\ge \mathbb{P}_r(s'' \le \hat{q}''_\alpha) - \frac{2Lc}{\pi_r} \\
&\ge 1 - \alpha - \frac{2Lc}{\pi_r},
\end{aligned}$$

where the last inequality follows from the conformal coverage guarantee for s'' (i.e., (5) and Lemma E.1).

Similarly, we have

$$\mathbb{E}F_{f,s'}(\hat{q}'_\alpha) - \mathbb{E}F_{f,s'}(\hat{q}'_\alpha - 2c) \le \frac{2Lc}{\pi_f}$$

and

$$\begin{aligned}
\mathbb{E}F_{f,s'}(\hat{q}'_\alpha) &\le \mathbb{E}F_{f,s'}(\hat{q}'_\alpha - 2c) + \frac{2Lc}{\pi_f} \\
&= \mathbb{P}_f(s' \le \hat{q}'_\alpha - 2c) + \frac{2Lc}{\pi_f} \\
&\le \mathbb{P}_f(s'' \le \hat{q}'_\alpha - c) + \frac{2Lc}{\pi_f} \\
&\le \mathbb{P}_f(s'' \le \hat{q}''_\alpha) + \frac{2Lc}{\pi_f} \\
&\le 1 - \left(\beta - \frac{2Lc}{\pi_f}\right),
\end{aligned}$$

where the last inequality follows from the conformal miscoverage guarantee for s'' (i.e., (6) and Lemma E.1). The proof is now complete. □

Proposition E.2. *Suppose the following:*

i) $F_{\ell,s}(\cdot)$ *is L-Lipschitz for* $\ell \in \{r, f\}$.

ii) $\mathbb{E}|s' - s| \le c$, *for a constant* $c \ge 0$.

iii) $\mathbb{E}|s| \le m$, *for a constant* $m \ge 0$.

Let $n = |\mathcal{D}_{\text{calib}}|$. *Then,*

$$\alpha' \le \alpha + 2\sqrt{Lc} + L(c + 2m), \tag{29}$$

$$\beta' \ge \beta - 2\sqrt{Lc} - L(c + 2m). \tag{30}$$

To prove Proposition E.2, we first show a preliminary lemma.

Lemma E.2. *Under the assumptions of Proposition E.2, we have for any t and $\epsilon > 0$,*

$$F_{\ell,s}(t-\epsilon) - \frac{c}{\epsilon} \le F_{\ell,s'}(t) \le F_{\ell,s}(t+\epsilon) + \frac{c}{\epsilon}, \tag{31}$$

for $\ell \in \{r, f\}$.

Proof. We have for any t and $\epsilon > 0$,

$$\begin{aligned} F_{\ell,s'}(t) &= \mathbb{P}_\ell(s' \le t) \\ &\le \mathbb{P}_\ell(s \le t+\epsilon) + \mathbb{P}_\ell(|s'-s| > \epsilon) \\ &\le F_{\ell,s}(t+\epsilon) + \frac{c}{\epsilon}, \end{aligned}$$

where the last inequality follows from the Markov inequality. A similar proof yields

$$F_{\ell,s'}(t) \ge F_{\ell,s}(t-\epsilon) - \frac{c}{\epsilon},$$

and the proof is complete. □

We are now ready to prove Proposition E.2.

Proof of Proposition E.2. We prove the bound (29); the proof of (30) is similar. For any $\epsilon > 0$,

$$\begin{aligned} &\mathbb{P}_r(s' \le \hat{q}'_\alpha) \\ &= \mathbb{E}\mathbb{P}_r(s' \le \hat{q}'_\alpha \,|\, \mathcal{D}_{\text{calib}}) \\ &\ge \mathbb{E}\mathbb{P}_r(s \le \hat{q}'_\alpha - \epsilon \,|\, \mathcal{D}_{\text{calib}}) - \frac{c}{\epsilon}, \\ &= \mathbb{E}[F_{r,s}(\hat{q}'_\alpha - \epsilon) - F_{r,s}(\hat{q}_\alpha) + F_{r,s}(\hat{q}_\alpha)] - \frac{c}{\epsilon}, \\ &\ge -L(\epsilon + \mathbb{E}|\hat{q}'_\alpha - \hat{q}_\alpha|) + \mathbb{E}F_{r,s}(\hat{q}_\alpha) - \frac{c}{\epsilon} \\ &\ge -L(\epsilon + c + 2\mathbb{E}|s|) + \mathbb{E}F_{r,s}(\hat{q}_\alpha) - \frac{c}{\epsilon} \\ &\ge 1 - (\alpha + L(\epsilon + c + 2m) + \frac{c}{\epsilon}), \end{aligned}$$

where the first inequality follows from Lemma E.2, and the last inequality follows from the conformal coverage guarantee for s (i.e., (5)). Maximizing the right-hand side over $\epsilon > 0$ gives the desired bound (29), and the proof is complete. □

By leveraging Lemma E.1, we observe that the conclusions of Proposition E.2 remain valid when s is substituted with $g(s)$, provided that $g(\cdot)$ is a strictly increasing function with bounded derivatives. The results in this section demonstrate that (α, β)-conformal unlearning exhibits robustness to bounded perturbations in the conformity score function. Specifically, when an alternative score s' is employed during inference and is statistically close to the original score s, the coverage and miscoverage guarantees degrade in a controlled manner. Consequently, the unlearning guarantees retain their statistical significance and interpretability, even when different conformity scores are utilized in downstream applications.

F Conformal Unlearning Beyond Exchangeability

The work (Foygel Barber et al., 2023) provides a framework to relax the exchangeability requirement and still obtain meaningful coverage bounds for CP. Specifically, when split CP is applied to a calibration set $\mathcal{D}_{\text{calib}}$ that is *not* exchangeable with $\mathcal{D}_{\text{test}}$, let $|\mathcal{D}_{\text{calib}}| = n$ and $\mathcal{D}_{\text{calib}} = \{Z_1, \ldots, Z_n\}$. The coverage bound becomes

$$\mathbb{P}(Y \in \mathcal{C}_\theta(X)) \ge 1 - \alpha - \sum_{i=1}^{n} \widetilde{\omega}_i \cdot d_{TV}\big(s(\boldsymbol{Z}), s(\boldsymbol{Z}^{\boldsymbol{i}})\big), \tag{32}$$

where $\widetilde{\omega}_i = \frac{\omega_i}{\omega_1+\cdots+\omega_n+1}$ for a set of user-specified weights $\{\omega_i\}_{i=1}^n$, $d_{TV}(P,Q)$ denotes the total variation distance between distributions P and Q, $s(\cdot)$ is the nonconformity score function, $\boldsymbol{Z} = \{Z_1,\ldots,Z_n,Z\}$, and $\boldsymbol{Z^i}$ is the set $\boldsymbol{Z}$ with the ith entry swapped with Z, i.e., $\boldsymbol{Z^i} = \{Z_1,\ldots,Z_{i-1},Z,Z_{i+1},\ldots,Z_n,Z_i\}$. Intuitively, the weights ω_i can be chosen so that the samples Z_i more similar to the test sample Z receive higher weight. For further details on non-exchangeable CP, see (Foygel Barber et al., 2023).

The primary concern in this context is the impact of the gap correction on our results, particularly in Lemma 1. To address this, we define the correction gap for a test sample Z as $g(Z)$. Under the assumption of non-exchangeability, the coverage guarantee is adjusted as follows:

$$\mathbb{P}(Y \in \mathcal{C}_\theta(X)) \geq 1-\alpha-g(Z). \tag{33}$$

It follows that for Lemma 1 to hold, we have to include the coverage gap into (5) and (6).

Proposition F.1. *In the conformal unlearning scenario, suppose $\pi_r > 0$. Moreover, suppose $g(Z) > 0$ (otherwise $\mathcal{D}_{\text{test}}$ and $\mathcal{D}_{\text{calib}}$ will be exchangeable). Then,* (6) *implies* (5) *if $\beta' \geq \alpha' \geq \alpha + g(Z)$, where α' and β' are to replace α and β in* (5) *and* (6)*, respectively.*

Proof. From (33), we have that $\mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) \geq 1-\alpha-g(Z)$. From (6), we obtain

$$\mathbb{P}_f(Y \in \mathcal{C}_{\theta_u}(X)) \leq 1-\beta' \leq 1-\alpha'.$$

Suppose (5) does not hold (i.e., $\mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) < 1-\alpha'$). Then,

$$\begin{aligned}\mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) &= \pi_r \mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) + \pi_f \mathbb{P}_f(Y \in \mathcal{C}_{\theta_u}(X)) \\ &< 1-\alpha',\end{aligned}$$

a contradiction if $\alpha' \geq \alpha + g(Z)$. Since $\beta' \geq \alpha'$ by definition, the proposition holds. □

Even in scenarios where exchangeability is difficult to assume, conformal unlearning remains a robust framework, provided that minor corrections are incorporated into the coverage and miscoverage bounds. This robustness ensures that the framework can accommodate practical situations where strict exchangeability is not guaranteed, while still delivering meaningful statistical guarantees. As demonstrated in Proposition F.1, the inclusion of a coverage gap correction enables the adaptation of conformal unlearning to non-exchangeable settings, thereby extending its applicability to real-world data distributions.

The lack of exchangeability often arises due to various types of distributional shifts. For instance, (Podkopaev & Ramdas, 2021) address shifts in label distributions, (Tibshirani et al., 2019) focus on covariate shifts in the input data, and (Chernozhukov et al., 2018) examine dependencies in time-series data. In general, as discussed earlier, the challenge lies in appropriately selecting weights to account for these distributional shifts. These weights are used to adjust the conformity scores of $\mathcal{D}_{\text{calib}}$ when determining the $(1-\alpha)$-th quantile. Developing practical conformal unlearning methods that effectively handle diverse types of distributional shifts represents an important avenue for future research, which is beyond the scope of this paper.

To illustrate the impact of affine transformations, we evaluate EFACE under mean and standard deviation shifts applied simultaneously on $\mathcal{D}_r$ and $\mathcal{D}_f$. Fig. 4 shows that $\mathsf{C}_{\mathcal{V}_r}$, $\mathsf{M}_{\mathcal{V}_f}$, and H exhibit minimal variation under the considered shifts (with an exception when std scaling < 1.0, where compression of the training distribution causes a train-validation mismatch: the model learns on data with reduced variance but is evaluated on validation data with the original, larger variance, leading to degraded overall performance). Furthermore, during the experiments, we observed no significant change in the accuracy across $\mathcal{V}_r$ and $\mathcal{V}_f$, and thus these results are omitted for brevity. These findings suggest that EFACE demonstrates resilience to small affine transformations, further underscoring its robustness in practical applications.

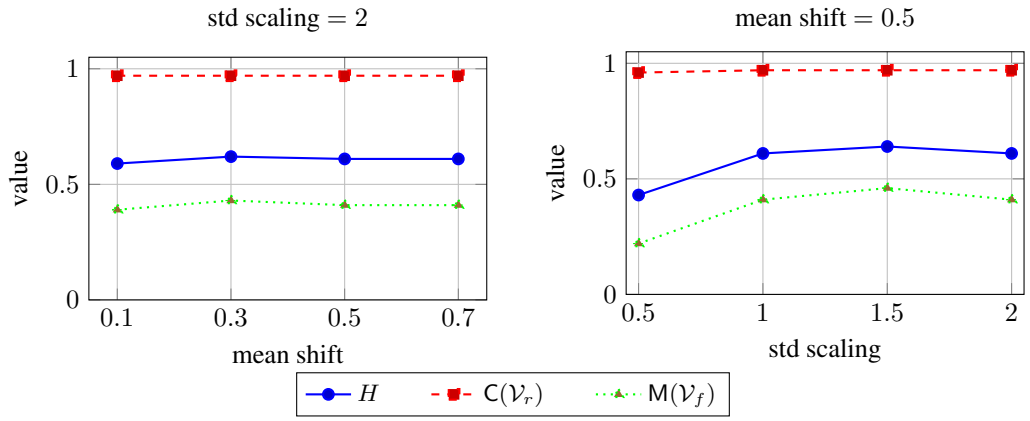


Figure 4: CIFAR100, RepVGG-a2 5-class forgetting with $\alpha = 0.05$ and $c = d = 100$ (EFACE). The shifts are applied after normalization. Left: metrics vs mean shift. Right: metrics vs std scaling.

G Proof of Main Text Theoretical Results

G.1 Proof of Lemma 1

From Assumption 1 and (4), we have for $(X, Y) \in \mathcal{D}_{\text{test}}$, $\mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) \geq 1 - \alpha$. From (6), we obtain

$$\mathbb{P}_f(Y \in \mathcal{C}_{\theta_u}(X)) \leq 1 - \beta \leq 1 - \alpha.$$

Suppose (5) does not hold (i.e., $\mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) < 1 - \alpha$). Then,

$$\begin{aligned}\mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) &= \mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X))\pi_r + \mathbb{P}_f(Y \in \mathcal{C}_{\theta_u}(X))\pi_f \\ &< 1 - \alpha,\end{aligned}$$

a contradiction, and the proof is complete.

G.2 Proof of Lemma 2

From (4), we have the marginal coverage guarantee: for $(X, Y) \in \mathcal{D}_{\text{test}}$,

$$\begin{aligned}1 - \alpha &\leq \mathbb{P}(Y \in \mathcal{C}_{\theta_u}(X)) \\ &= \pi_r \mathbb{P}_r(Y \in \mathcal{C}_{\theta_u}(X)) + \pi_f \mathbb{P}_f(Y \in \mathcal{C}_{\theta_u}(X)) \\ &\leq 1 - \pi_f + \pi_f(1 - \beta) = 1 - \pi_f \beta,\end{aligned}$$

where the last inequality follows from (6). Rearranging the terms gives the desired result.

G.3 Proof of Proposition 1

From (6), we have

$$\begin{aligned}\beta &\leq \mathbb{P}(s(X_f, Y_f; \theta_u) > \hat{q}_\alpha) \\ &\leq \mathbb{P}(s(X_f, Y_f; \theta_u) \geq s(X_r, Y_r; \theta_u) \,|\, s(X_r, Y_r; \theta_u) \leq \hat{q}_\alpha) \\ &\leq \frac{\mathbb{P}(s(X_f, Y_f; \theta_u) \geq s(X_r, Y_r; \theta_u))}{\mathbb{P}(s(X_r, Y_r; \theta_u) \leq \hat{q}_\alpha)} \\ &\leq \frac{\mathbb{P}(s(X_f, Y_f; \theta_u) \geq s(X_r, Y_r; \theta_u))}{1 - \alpha},\end{aligned}$$

where the first inequality follows from (6), the second inequality holds due to independence of $\mathcal{V}_f$ from $\mathcal{D}_{\text{calib}}, \mathcal{V}_r$, and the last inequality follows from (5). Therefore, the result holds.

G.4 Proof of Corollary 1

Define the following events:

$$A = \{Y \in \mathcal{C}_{\theta_u}(X)\},$$
$$Q = \{|\mathcal{C}_{\theta_u}(X)| \leq c\},$$
$$M = \{|\mathcal{C}_{\theta_u}(X)| \leq d\}.$$

From (5), we have

$$1 - \alpha \leq \mathbb{P}_r(A \,|\, Q). \tag{34}$$

On the other hand, by the law of total probability, we have

$$\begin{aligned}\mathbb{P}_r(A \,|\, Q) &= \mathbb{P}_r(A \,|\, Q)\mathbb{P}_r(Q) + \mathbb{P}_r(A \,|\, Q^{\mathsf{c}})\mathbb{P}_r(Q^{\mathsf{c}}) \\ &\leq \mathbb{P}_r(A \,|\, Q) + \mathbb{P}_r(Q^{\mathsf{c}}) \\ &\leq \mathbb{P}_r(A \,|\, Q) + \zeta_c. \end{aligned} \tag{35}$$

Combining (34) and (35), we have

$$\mathbb{P}_r(A \,|\, Q) \geq 1 - \alpha - \zeta_c. \tag{36}$$

Furthermore, from (6), we have

$$\beta \leq \mathbb{P}_f(A^{\mathsf{c}}). \tag{37}$$

Again, by the law of total probability, we have

$$\begin{aligned}\mathbb{P}_f(A^{\mathsf{c}}) &= \mathbb{P}_f(A^{\mathsf{c}} \,|\, M)\mathbb{P}_f(M) + \mathbb{P}_f(A^{\mathsf{c}} \,|\, M^{\mathsf{c}})\mathbb{P}_f(M^{\mathsf{c}}) \\ &\leq \mathbb{P}_f(A^{\mathsf{c}} \,|\, M) + \mathbb{P}_f(M^{\mathsf{c}}) \\ &\leq \mathbb{P}_f(A^{\mathsf{c}} \,|\, M) + \eta_d. \end{aligned} \tag{38}$$

Combining (37) and (38), we have

$$\mathbb{P}_f(A^{\mathsf{c}} \,|\, M) \geq \beta - \eta_d. \tag{39}$$

The proof is now complete.

H More on the Datasets and Baselines

H.1 Datasets

CIFAR100 (Krizhevsky, 2009) is a labeled subset of the 80 Million Tiny Images dataset. It comprises 60,000 color images of size 32×32 pixels, evenly distributed across 100 object classes, so each class contains exactly 600 images, split into 500 for training and 100 for testing. The 100 classes are further organized into 20 higher-level "superclasses", and every image therefore carries two annotations: a fine label denoting its class and a coarse label denoting its superclass. We use the fine labels, and we load the dataset using torchvision's datasets library.

ImageNet100 (Shekhar, 2021) is a subset of ILSVRC 2012 containing 100 classes randomly sampled from the original 1,000, and the included categories are listed in a `Labels.json` file. It is commonly used as a smaller proxy for ImageNet.

20NewsGroups (Lang, 1995) comprises roughly 20,000 English posts nearly evenly distributed across 20 topics, and it is a standard benchmark for text classification and clustering. Documents are plain text, consisting of headers plus a message body. We use the SetFit release, which stores `text`, an integer `label` in $\{0, \ldots, 19\}$, and `label_text`.

H.2 On The Baselines

For our baseline comparison, we use a variety of baselines that constitute different approaches in the literature. Moreover, we include a baseline that directly targets a form of forget-data-miscoverage optimization.

- *Fine-tuning with a conformal miscoverage objective*: CPU (Shi et al., 2025), which fine-tunes while updating the conformity scores of the forget data.
- *Entropy maximization on forgotten data plus retained-data regularization*: BADT (Chundawat et al., 2023a), which drives the student toward a randomly initialized teacher on the forget data, that is toward maximal uncertainty, while distilling the original model on the retained data. SCRUB (Kurmanji et al., 2023) implements the same trade-off with a KL divergence that is maximized on the forget data and minimized on the retained data.
- *Negative-label or random-label training on forgotten groups*: AMN (Graves et al., 2021), which relabels the forgotten groups at random and then fine-tunes, $\nabla\tau$ which fine-tunes on the retained data while regularizing the loss such that the forgotten data are treated as unseen, and SalUn (Fan et al., 2024), which restricts random-label fine-tuning to the weights most salient to the forgotten data.
- *Decision-boundary shifting on the forgotten data only*: BS and BE (Chen et al., 2023), which relabel the forgotten data with their nearest incorrect class (BS) or with an extra shadow class that is pruned afterwards (BE).
- *Class-removal or concept-suppression baselines*: UNSIR (Tarun et al., 2024), which constructs loss-maximizing noise for the target class in an impair-repair scheme, and SSD (Foster et al., 2024), which dampens the parameters that are specialized for the forget data.
- *Certified and retraining references*: PABI (Koloskova et al., 2025) and the RT model.

We summarize the objectives of each of the baselines and how they achieve unlearning, except for CPU, which is the closest method to ours and is therefore discussed together with the related work in Appendix A. This section is not meant to be comprehensive but to give a clearer idea about the methods we compare with.

$\nabla\tau$ (Trippa et al., 2024) introduces a new loss objective that focuses on pushing the loss of the data meant to be forgotten from the training data to become larger than the loss of some validation data (carrying the same unlearned labels). They merge it with the original objective of minimizing the loss on the retained data. The new loss is then given by

$$L = \alpha\big(\mathrm{ReLU}(L_{\mathcal{D}_v} - L_{\mathcal{D}_f})\big) + (1-\alpha)L_{\mathcal{D}_r},$$

where α controls how much emphasis should be given to the retained versus forgotten data. In our framework, however, none of the subsets $\mathcal{D}_v$, $\mathcal{D}_f$, and $\mathcal{D}_r$ is seen during training. We assume that this causes the performance of $\nabla\tau$ to drop.

SCRUB (Kurmanji et al., 2023) builds the loss function on the KL-divergence of the unlearning model to a teacher that was trained on the full training data (both $\mathcal{D}_r$ and $\mathcal{D}_f$). This is nothing but the original base model trained on $\mathcal{D}_{\mathrm{train}}$. They add one more regular loss term to be minimized over the retained data to maintain performance on those points. The final loss that should be minimized becomes

$$L = \frac{\alpha}{N_r}\sum_{x_r\in\mathcal{D}_r} d_{KL}(x_r;\omega^u) + \frac{\gamma}{N_r}\sum_{(x_r,y_r)\in\mathcal{D}_r} l(f(x_r;\omega^u),y_r) - \frac{1}{N_f}\sum_{x_f\in\mathcal{D}_f} d_{KL}(x_f;\omega^u),$$

where N_r is the number of data points to retain, N_f the number of data points to forget, and α and γ control the importance of the terms of retaining. Notice that SCRUB tries to make the distributions of the unlearning model and the base model converge to each other on the retained data and diverge from each other on the forgotten data. Note that when $\mathcal{D}_{\mathrm{unlearn}}$ is a proxy set not used in pretraining, then the base model is less capable of correctly classifying $\mathcal{D}_r$, which leaves the unlearned model with high variance.

SSD (Foster et al., 2024) uses synaptic dampening of the parameters (weights) of the model which are "specialized" for $\mathcal{D}_f$. SSD compares the "importances" of the weights using the first-order derivative property of the Fisher Information Matrix (FIM) and decides whether to dampen a weight if it is more specialized for $\mathcal{D}_f$ than for other training data, as follows.

$$[]_{\mathcal{D}} = \mathbb{E}\left[- \frac{\partial^2 \ln p(\mathcal{D} \mid \theta)}{\partial \theta^2}\Big|_{\theta^*_D} \right],$$
$$[]_{\mathcal{D}} = \mathbb{E}\left[(\frac{\partial \ln p(\mathcal{D} \mid \theta)}{\partial \theta})(\frac{\partial \ln p(\mathcal{D} \mid \theta)}{\partial \theta})^T|_{\theta^*_D} \right].$$

$$\beta = \min(\lambda \frac{[]_{\mathcal{D},i}}{[]_{\mathcal{D}_{f,i}}}, 1),$$
$$\theta_i = \begin{cases} \beta\, \theta_i, & \text{if } []_{\mathcal{D}_{f,i}} > \alpha\, []_{\mathcal{D},i}, \\ \theta_i, & \text{if } []_{\mathcal{D}_{f,i}} \leq \alpha\, []_{\mathcal{D},i}, \end{cases} \forall\ i \in [0, |\theta|].$$

Generally, they assume that the training data importances can be calculated before training and then the importances of the parameters will be compared between $\mathcal{D}_f$ and $\mathcal{D}_{\text{train}}$. In our framework, $\mathcal{D}_f$ might not be part of $\mathcal{D}_{\text{train}}$. Hence, it is not straightforward to argue for a stable relation between the importances. That seems to be the reason why SSD fails to unlearn when $\mathcal{D}_{\text{unlearn}}$ is a proxy set.

AMN (Graves et al., 2021) randomly relabels the data to be forgotten. It replaces the classes to be forgotten with new random labels over the whole training set and then retrains the model for a few iterations over the newly labeled data. However, in our case we feed $\mathcal{D}_f$ and $\mathcal{D}_r$ to the unlearning algorithm rather than the full training set. Since $\mathcal{D}_f$ in our framework might not be used for training or is just a smaller subset compared to the fraction of forgotten data in the training set, we find that AMN overfits to $\mathcal{D}_{\text{unlearn}}$ and does not perform well on the other subsets of data.

BADT (Chundawat et al., 2023a) introduces a bad teacher initialized with random noise which induces forgetting by minimizing the KL-divergence between its distribution and that of the unlearned model (student) on the forgotten data. On the other hand, BADT minimizes the divergence between the distribution of the base model and that of the student on the retained data. The objective of BADT is given below.

$$L(x, l_u) = (1 - l_u)\mathcal{KL}(T_s(x)||S(x)) + l_u\mathcal{KL}(T_d(x)||S(x)),$$

where l_u is the label to be forgotten, x is a sample point, T_s is the base model, T_d is the bad teacher model, and $\mathcal{KL}(P, Q)$ is the KL-divergence between the P and Q. Note that BADT is initially proposed for label-wise forgetting but can be used in the cluster-wise case.

UNSIR (Tarun et al., 2024) constructs noisy data by maximizing the loss on the noisy samples that carry the label to be forgotten. Then, it feeds the loss-maximizing noise to the model along with some retrained data in an impair-repair fashion. UNSIR depends on the label to be forgotten to construct the loss-maximizing noise. Hence, it is not suitable for targeted cluster-wise forgetting. That is why we do not include its results in those scenarios.

SalUn (Fan et al., 2024) computes the saliency mask $m_S = \mathbf{1}\{|\nabla_\theta \ell_{CE}(\theta; \mathcal{D}_f)|_{\theta=\theta_o}| \geq \gamma\}$ at the original weights θ_o, where γ keeps a fixed fraction of the weights, and updates only the salient weights, $\theta_u = m_S \odot (\Delta\theta + \theta_o) + (1 - m_S) \odot \theta_o$, by minimizing the cross-entropy on $\mathcal{D}_f$ with random labels $y' \neq y$ plus α times the cross-entropy on $\mathcal{D}_r$. We include it as a recent and strong representative of random-label unlearning.

Boundary unlearning (Chen et al., 2023) shifts the decision boundary of the original model f_{θ_o} using $\mathcal{D}_f$ only. BS relabels each forget point with its nearest incorrect class, $y_{nbi} = \arg\max_k f^k_{\theta_o}(x + \epsilon\, \text{sign}(\nabla_x \ell(x, y; \theta_o)))$, and fine-tunes on the pairs (x, y_{nbi}), whereas BE adds a shadow class as an extra output neuron, fine-tunes on $\mathcal{D}_f$ relabeled with that class, and then prunes the neuron. Neither method depends on the forgotten label, so both apply to cluster-wise forgetting. For 20NewsGroups, the sign step of BS is taken in the token-embedding space. We include them because they unlearn in the output (decision) space, which is also the space targeted by conformal unlearning.

PABI (Koloskova et al., 2025) is an (ϵ,δ)-unlearning certified method that attempts to approximate a model trained without the forgotten data in the parameters space. In particular, they define (ϵ,δ)-unlearning as follows.

Definition 2 ((ϵ,δ)-unlearning (Koloskova et al., 2025, Def. 2.1)). *Let $\epsilon \geq 0$, $\delta \in [0, 1]$. We say that $\mathfrak{U}$ is (ϵ, δ)-unlearning algorithm for $\mathfrak{A}$ if there exists a certifying algorithm $\tilde{\mathfrak{A}}$, such that for any forget and initial datasets $\mathcal{D}_f \in \mathcal{D}$ and any observation $\theta \in \mathbb{R}^d$,*

$$\mathbb{P}(\mathfrak{U}(\mathfrak{A}(\mathcal{D}), \mathcal{D}, \mathcal{D}_f) = \theta) \leq e^{\epsilon}\mathbb{P}\Big(\tilde{\mathfrak{A}}(\mathcal{D} \setminus \mathcal{D}_f) = \theta\Big) + \delta,$$
$$\mathbb{P}\Big(\tilde{\mathfrak{A}}(\mathcal{D} \setminus \mathcal{D}_f) = \theta\Big) \leq e^{\epsilon}\mathbb{P}(\mathfrak{U}(\mathfrak{A}(\mathcal{D}), \mathcal{D}, \mathcal{D}_f) = \theta) + \delta.$$

Note that the certifying algorithm $\tilde{\mathfrak{A}}$ might not be the original training algorithm. It is just a training algorithm that is not trained on the forgotten data. Hence, its result might be a model very different that a model trained from scratch using the same original training algorithm but only on the retained data (RT). PABI induces this unlearning by adding noise to the gradients during training and clipping the weights. The exact approach is as follows.

$$x_0 = \prod_{C_0}(\hat{x}),$$
$$x_{t+1} = x_t - \gamma(\prod_{C_1}(g_t) + \lambda x_t) + \varsigma_{t+1},$$

where $\varsigma_{t+1} \sim \mathcal{N}(0, \sigma^2 I^d)$ is Gaussian noise, and $\prod_{C_0}$,$\prod_{C_1}$ are the clipping operators of radii $C_0, C_1 > 0$, respectively. The PABI method employs a combination of noisy updates to induce forgetting of $\mathcal{D}_f$, followed by fine-tuning on $\mathcal{D}_r$. In our experiments, the number of noisy update steps was minimal (typically one), resulting in a fine-tuning phase with a number of epochs equivalent to that of retraining from scratch. We observe that PABI does not achieve convergence to the same validation accuracy in fewer epochs relative to RT.

I Further Experimental Details

I.1 Reproducibility Details

This appendix collects the full experimental protocol in one place, namely the datasets and models, the computing environment and training recipes, the data partitions and calibration, the unlearning setup, the baselines, the evaluation metrics, and the hyperparameters used in every experiment.

Datasets and models. We evaluate on ImageNet100 (Shekhar, 2021), CIFAR100 (Krizhevsky, 2009), and 20NewsGroups (Lang, 1995), which are described in Appendix H.1. The main text reports ImageNet100, and Appendix I.2 adds CIFAR100 and 20NewsGroups. For the image datasets, the model before unlearning is RepVGG-A2, loaded with pretrained weights for CIFAR100[2] and ImageNet[3], except in the comparisons with RT, PABI, and CPU (Tables 3, 4 and 11 to 13), which use a ResNet18 that we train from scratch with the recipes below. For 20NewsGroups, the model is DistilBERT (`distilbert-base-uncased`). Unless otherwise specified, all results are averaged over six random seeds, except for the RT model and the certified method (PABI), which are averaged over three seeds due to their significant computational overhead, and the 20NewsGroups experiments, which are also averaged over three seeds. In all cases, we observe minimal variability across seeds.

Environment. Four NVIDIA RTX A5000 GPUs; PyTorch with `nn.DataParallel`; batch size 256; dataloader workers 2; no memory pinning. All vision inputs are normalized with the standard dataset statistics and use standard data augmentation, and text tokenization follows the DistilBERT pipeline from the `transformers` library.

Training recipes. CIFAR100: ResNet18, SGD 50 epochs, initial lr 0.1 with linear decay to 10^{-4}, momentum 0.9, wd 5×10^{-4}, cross-entropy, no early stopping. **ImageNet100 (100 labels):** ResNet18, same as CIFAR100 but 80 epochs. **20NewsGroups:** DistilBERT (`distilbert-base-uncased`), 15 epochs, initial lr 0.01, otherwise as above.

[2]Provided here: https://github.com/chenyaofo/pytorch-cifar-models

[3]Provided here: https://github.com/DingXiaoH/RepVGG/blob/main/repvgg.py

Data partitions and calibration. The experimental setup involves six data subsets, namely the training forget/retain splits $(\mathcal{T}_f, \mathcal{T}_r)$, the unlearning forget/retain splits $(\mathcal{D}_f, \mathcal{D}_r)$, and the unseen forget/retain splits $(\mathcal{V}_f, \mathcal{V}_r)$, together with the calibration set $\mathcal{D}_{\text{calib}}$. Concretely, 90% of the train split is used for training through $\mathcal{D}_{\text{train}}$, and the remaining 10% of the train split is used for validation, if any baseline (such as $\nabla\tau$) uses a validation split internally (5%), and for $\mathcal{V}_r$ and $\mathcal{V}_f$ (5%). Moreover, 80% of the test split is used for the (reconformalization) calibration set $\mathcal{D}_{\text{calib}}$, and 20% is used for $\mathcal{D}_r$ and $\mathcal{D}_f$. Hence, we differentiate between the following splits.

(i) the data used to optimize the unlearned model, $\mathcal{D}_{\text{unlearn}} = \mathcal{D}_r \cup \mathcal{D}_f$;

(ii) the data used for conformal calibration (reconformalization), $\mathcal{D}_{\text{calib}}$;

(iii) the data used for evaluation, $\mathcal{D}_{\text{test}} = \mathcal{V}_r \cup \mathcal{V}_f$, which are held out and are used neither for pretraining nor for unlearning;

(iv) the pretraining splits $\mathcal{T}_r$ and $\mathcal{T}_f$.

All of these are drawn from the same semantic groups under test, and $\mathcal{D}_{\text{calib}}$ and $(\mathcal{V}_r \cup \mathcal{V}_f)$ are exchangeable at all times.

The proposed framework applies unlearning to $\mathcal{D}_f$ and $\mathcal{D}_r$. When these sets are disjoint from the training data, the scenario is referred to as *out-sample unlearning* (*Out*). Conversely, when $\mathcal{D}_f$ and $\mathcal{D}_r$ are subsets of the training data $(\mathcal{T}_f, \mathcal{T}_r)$, so that $\mathcal{D}_{\text{unlearn}} \subset (\mathcal{T}_r \cup \mathcal{T}_f)$, which is the common scenario in the machine unlearning literature, the scenario is termed *in-sample unlearning* (*In*). Results for both scenarios are presented in the corresponding tables.

In all experiments, we perform reconformalization for all methods using $\mathcal{D}_{\text{calib}}$ to obtain valid unlearned CP sets after the unlearning is completed. The numbers we report on $\mathcal{T}_r$, $\mathcal{T}_f$, $\mathcal{D}_r$, and $\mathcal{D}_f$ are in-sample diagnostics. They are reported for completeness, in order to show that even points used in the original training or in the unlearning step are no longer covered by the prediction sets, which is of practical interest. The distribution-free guarantee of (4) is claimed only for the held-out evaluation data $\mathcal{V}_r$ and $\mathcal{V}_f$, which satisfy the exchangeability condition with respect to $\mathcal{D}_{\text{calib}}$.

Cluster-wise unlearning. In the cluster-wise experiments, we partition the training data into k clusters within the embedding space using the k-means clustering algorithm, where $k = |\mathcal{Y}|$ corresponds to the number of classes. That is, we take the representation of the points produced by the pretrained model (the same model used as the original model prior to unlearning) and perform k-means in that space. Each data point is then assigned a pseudo-label based on its proximity to the nearest cluster centroid, and the forget data are identified based on these pseudo-labels by picking a specified number of clusters to forget. This ensures that the data designated for unlearning share common high-level characteristics, which aligns with the foundational principles of conformal unlearning, as outlined in Section 2. Results for both cluster-wise and class-wise unlearning are provided in Appendix I.2.

Baselines. We evaluate our approach against several state-of-the-art unlearning methods, including $\nabla\tau$ (Trippa et al., 2024), AMN (Graves et al., 2021), SCRUB (Kurmanji et al., 2023), BADT (Chundawat et al., 2023a), SSD (Foster et al., 2024), UNSIR (Tarun et al., 2024), SalUn (Fan et al., 2024), BS and BE (Chen et al., 2023), and CPU (Shi et al., 2025), which constitute a variety of unlearning methods spanning several approaches, and we explain these methods in Appendix H.2. We also compare against the RT baseline applied to $\mathcal{T}_r$ and the certified unlearning method PABI (Koloskova et al., 2025), which also operates exclusively on $\mathcal{T}_r$. We use the authors' repositories (from (Foster et al., 2024; Chundawat et al., 2023a)), and we implement PABI on our own. We grid-search around the released settings to keep compute comparable only for CIFAR100 and ImageNet100, and we use those same settings for 20NewsGroups. We reimplement SalUn, BS, and BE following the authors' code and tune them per dataset in the in-sample setting, selecting the largest H over $\mathcal{D}_r$, $\mathcal{D}_f$, $\mathcal{V}_r$, and $\mathcal{V}_f$ among the settings whose $A(\mathcal{V}_r)$ is within 8 points of the original model, with at most 50 epochs (15 for 20NewsGroups). Our evaluation applies unlearning on $\mathcal{D}_f/\mathcal{D}_r$.

Unlearning optimization. For EFACE, we retain SGD with the base momentum and weight decay, we tune only the learning rate (grid-search), and we use no scheduler during unlearning. For the other baselines, we keep the same optimizers used in the authors' repositories, and for RT and PABI, the original training algorithm and hyperparameter setup are used for fine-tuning.

Evaluation metrics. We report $\mathsf{C}_{\mathcal{D}}(c)$, the coverage frequency (CF) on the retained subsets $\mathcal{D} \in \{\mathcal{D}_r, \mathcal{T}_r, \mathcal{V}_r\}$, and $\mathsf{M}_{\mathcal{D}}(d)$, the miscoverage frequency (mCF) on the forgotten subsets $\mathcal{D} \in \{\mathcal{D}_f, \mathcal{T}_f, \mathcal{V}_f\}$, with c=d, together with the harmonic mean H of these six conformal metrics x_i, defined as $H = \frac{6}{\sum_i x_i^{-1}}$ with H=0 if any x_i=0. Furthermore, we report the accuracy $A_{\mathcal{D}}$ for each subset $\mathcal{D} \in \{\mathcal{D}_r, \mathcal{D}_f, \mathcal{T}_r, \mathcal{T}_f, \mathcal{V}_r, \mathcal{V}_f\}$ (before the conformalization) to further illustrate the phenomenon of fake conformal unlearning. Additional metrics include the Membership Inference Attack (MIA) Difference, described next, and the unlearning time, denoted as Tsec and measured in seconds. For the RT and PABI methods, which perform unlearning on the entirety of the retained dataset $\mathcal{T}_r$, we omit results for the subsets $\mathcal{D}_r \subset \mathcal{T}_r$ and $\mathcal{D}_f \subset \mathcal{T}_f$. Implementation details for C/M and the calibration protocol follow the main text theory.

MIA evaluation. For each sample we extract: loss, entropy, prediction margin, logit ℓ_2-norm, and top-k probabilities (dynamic k). We train a RandomForest attacker with stratified 10-fold cross-validation and report the MIA Difference, also known as the *Adversarial Advantage*, that is, the attacker's accuracy percentage minus the majority-class ratio, so an optimal value approaches zero.

Code and hyperparameters. Scripts for dataset preparation, partition seeds, hyperparameter grids, and exact command lines are provided in the accompanying repository (to be made public upon publication). We fix seeds for data splits and model initialization to ensure full reproducibility. The hyperparameters used in our experiments are detailed below.

- **EFACE:** $\kappa = 5.0$ and $\gamma = 0.0$ in all experiments. Number of epochs is 20, except for CIFAR100 cluster-wise forgetting (35 epochs). Learning rate is 0.04 for all experiments, except for ImageNet100 RepVGG-A2 in-sample forgetting ($\mathrm{lr} = 0.02$) and Table 4 ($\mathrm{lr} = 0.08$).
- $\nabla\tau$**:** Split ratio $= 0.5$ and $\mathrm{lr} = 1 \times 10^{-4}$ in all experiments.
- **SCRUB:** Epochs $= 10$, $\gamma = 2.0$, $\beta = 0.1$, msteps $= 3$, sstart $= 10$, kd-T $= 4$, and $\mathrm{lr} = 0.01$ in all experiments, except for ImageNet100 in-sample class-wise forgetting ($\mathrm{lr} = 0.04$).
- **AMN:** $\mathrm{lr} = 1 \times 10^{-3}$ and epochs $= 8$ in all experiments.
- **SSD:** $\lambda = 1$ in all experiments. $\alpha = 20$ for in-sample experiments and $\alpha = 50000$ for out-of-sample experiments (larger values cause SSD to collapse in the out-of-sample setting).
- **UNSIR:** 150 noise-generating epochs, 5 impair epochs, and 10 repair epochs, with $\mathrm{lr} = 1 \times 10^{-4}$.
- **BADT:** KL temperature $= 1.0$, $\mathrm{lr} = 0.03$, and epochs $= 10$ in all experiments.
- **SalUn:** Mask ratio $= 0.7$, $\alpha = 1$, and epochs $= 10$ in all experiments, with $\mathrm{lr} = 0.05$ for ImageNet100, 0.01 for CIFAR100, and 0.03 for 20NewsGroups.
- **BS:** $\epsilon = 0.01$ in all experiments, with (lr, epochs) $= (3{\times}10^{-3}, 50)$ for ImageNet100, $(10^{-3}, 15)$ for CIFAR100, and $(3{\times}10^{-4}, 15)$ for 20NewsGroups.
- **BE:** (lr, epochs) $= (10^{-3}, 50)$ for ImageNet100, $(10^{-4}, 15)$ for CIFAR100, and $(3{\times}10^{-4}, 5)$ for 20NewsGroups.
- **PABI** (used in Table 3): Constant noise scheduler, $\epsilon = 1.0$, $\delta = 1 \times 10^{-5}$, initial model clipping constant $= 0.01$, gradient clipping constant $= 10.0$, maximum learning rate $= 0.001$, $\lambda = 500$, and initial $\sigma = 0.0$.

I.2 More Results

In this appendix we show more results on ImageNet100, CIFAR100 and 20NewsGroups for both class-wise and cluster-wise unlearning, where cluster-wise unlearning is performed as described in Appendix I.1. For the image datasets, the model before unlearning is RepVGG-a2, while for the text dataset, it is DistilBERT. We also demonstrate more results over ImageNet100 and CIFAR100 for the comparison of EFACE with RT and PABI. Note that PABI is the only certified unlearning method we are aware of that does not assume convexity of the loss function, and hence suitable for our experiments. The results presented here are consistent with Table 3: RT and PABI demonstrate *fake conformal unlearning* in the CIFAR100 cluster-wise unlearning case also.

In all the tables containing our proposed metrics, we follow the same convention as in the main text by highlighting the best, second best, and third best M and H results with **red**, **blue**, and **cyan**, respectively. Note that by Lemma 2, β is subject to the bound $\beta \leq \frac{\alpha}{\pi_f}$. E.g., if $\alpha = 0.05$ (as is the case in the following tables), and $\pi_f \approx 0.10$, then $\beta \leq 0.50$, marginally over the forget data subsets.

I.2.1 Class-wise forgetting on ImageNet100

Table 5 presents the results for the class-wise forgetting scenario for ImageNet100. All methods achieve better conformal unlearning performance compared to cluster-wise forgetting, likely due to the clearer separation between the forget and retain data (belonging to distinct classes) along the model's decision boundaries. Note that in this case $\nabla\tau$, SalUn, SSD, and BADT suffer a big drop in unlearning performance when moving from the in-sample case to the out-sample case. EFACE consistently demonstrates superior performance compared to all competing methods across all data subsets and in both in-sample and out-sample scenarios. The significant margin achieved by EFACE (≈ 0.19 in-sample) underscores its robustness and efficacy in simultaneously achieving high retained coverage and substantial forget miscoverage.

Table 5: ImageNet100, RepVGG-a2 *class*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten classes. Coverage/miscoverage results.

Split	Method	$\mathsf{C}_c(\mathcal{D}_r) \uparrow$	$\mathsf{M}_d(\mathcal{D}_f) \uparrow$	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$
	OR	1.00 ± 0.00	0.01 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.01 ± 0.00	0.01 ± 0.00
In	$\nabla\tau$	0.83 ± 0.37	0.46 ± 0.24	0.83 ± 0.37	0.40 ± 0.22	0.83 ± 0.37	0.38 ± 0.21	0.55 ± 0.12
	AMN	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00	0.37 ± 0.01	0.99 ± 0.00	0.36 ± 0.02	0.63 ± 0.01
	SalUn	1.00 ± 0.00	0.58 ± 0.42	0.99 ± 0.00	0.67 ± 0.08	0.98 ± 0.00	0.71 ± 0.10	0.78 ± 0.13
	SCRUB	0.86 ± 0.19	0.76 ± 0.27	0.85 ± 0.21	0.78 ± 0.25	0.84 ± 0.21	0.80 ± 0.23	0.81 ± 0.10
	BADT	0.99 ± 0.00	0.13 ± 0.01	0.99 ± 0.00	0.14 ± 0.01	0.98 ± 0.00	0.19 ± 0.02	0.26 ± 0.01
	SSD	0.99 ± 0.00	0.47 ± 0.03	0.99 ± 0.00	0.40 ± 0.03	0.99 ± 0.00	0.39 ± 0.02	0.59 ± 0.02
	UNSIR	1.00 ± 0.00	0.13 ± 0.01	0.99 ± 0.00	0.14 ± 0.01	0.98 ± 0.00	0.14 ± 0.01	0.24 ± 0.01
	BS	1.00 ± 0.00	0.94 ± 0.00	1.00 ± 0.00	0.55 ± 0.00	0.99 ± 0.00	0.55 ± 0.00	0.78 ± 0.00
	BE	1.00 ± 0.00	0.74 ± 0.04	1.00 ± 0.00	0.65 ± 0.03	0.99 ± 0.00	0.64 ± 0.02	0.80 ± 0.01
	CPU	1.00 ± 0.00	0.41 ± 0.01	0.99 ± 0.00	0.37 ± 0.00	0.99 ± 0.00	0.36 ± 0.01	0.55 ± 0.00
	EFACE	1.00 ± 0.00	0.99 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
Out	$\nabla\tau$	1.00 ± 0.00	0.14 ± 0.05	0.99 ± 0.00	0.04 ± 0.02	0.98 ± 0.00	0.05 ± 0.02	0.11 ± 0.03
	AMN	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00	0.14 ± 0.02	0.98 ± 0.00	0.15 ± 0.03	0.33 ± 0.03
	SalUn	1.00 ± 0.00	0.00 ± 0.00	0.99 ± 0.00	0.68 ± 0.04	0.99 ± 0.00	0.65 ± 0.03	0.00 ± 0.00
	SCRUB	1.00 ± 0.00	0.97 ± 0.03	1.00 ± 0.00	0.65 ± 0.01	0.99 ± 0.00	0.70 ± 0.01	0.86 ± 0.01
	BADT	0.96 ± 0.00	0.12 ± 0.01	1.00 ± 0.00	0.02 ± 0.00	0.98 ± 0.00	0.03 ± 0.00	0.06 ± 0.00
	SSD	0.96 ± 0.00	0.05 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.98 ± 0.00	0.01 ± 0.00	0.02 ± 0.00
	UNSIR	1.00 ± 0.00	0.19 ± 0.03	0.99 ± 0.00	0.15 ± 0.01	0.98 ± 0.00	0.13 ± 0.01	0.26 ± 0.01
	BS	0.98 ± 0.00	0.98 ± 0.00	1.00 ± 0.00	0.45 ± 0.00	0.99 ± 0.00	0.49 ± 0.00	0.72 ± 0.00
	BE	0.96 ± 0.00	0.86 ± 0.04	0.98 ± 0.01	0.20 ± 0.04	0.98 ± 0.00	0.19 ± 0.03	0.42 ± 0.03
	CPU	1.00 ± 0.00	0.31 ± 0.01	0.99 ± 0.00	0.18 ± 0.01	0.98 ± 0.00	0.19 ± 0.01	0.35 ± 0.01
	EFACE	0.99 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.79 ± 0.01	1.00 ± 0.00	0.82 ± 0.00	0.92 ± 0.00

I.2.2 Accuracy results for ImageNet100 cluster-wise forgetting

Table 6 reports the accuracy results for ImageNet100 cluster-wise forgetting, which underlie the discussion of fake conformal unlearning in Section 5.1.

I.2.3 MIA and unlearning time

In Table 7, an MIA Diff value close to zero is the desired outcome, and all methods, including EFACE, score low. An adversary employing the MIA technique outlined in Appendix I.1 would therefore find it difficult to distinguish between forgotten data points used during pretraining ($\mathcal{T}_f$) and unseen data points ($\mathcal{V}_f$). In terms of unlearning time efficiency, EFACE is at the slower side (but not the slowest) due to the quantile calculation step at each iteration which requires a full forward-pass of the data to find the scores and their quantile.

Table 6: ImageNet100, RepVGG-a2 *cluster*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten clusters. Accuracy results in %.

Split	Method	$A(\mathcal{D}_r) \uparrow$	$A(\mathcal{D}_f) \downarrow$	$A(\mathcal{T}_r) \uparrow$	$A(\mathcal{T}_f) \downarrow$	$A(\mathcal{V}_r) \uparrow$	$A(\mathcal{V}_f) \downarrow$
	OR	96.13 ± 0.00	94.97 ± 0.00	96.16 ± 0.00	93.32 ± 0.00	91.52 ± 0.00	88.35 ± 0.00
In	$\nabla\tau$	90.57 ± 7.09	24.15 ± 4.35	81.91 ± 5.59	26.31 ± 4.87	79.76 ± 5.11	26.50 ± 4.73
	AMN	99.98 ± 0.00	0.00 ± 0.00	92.06 ± 0.06	28.15 ± 1.39	88.85 ± 0.13	27.02 ± 1.36
	SalUn	99.99 ± 0.00	2.45 ± 1.52	86.65 ± 0.21	14.96 ± 1.42	84.60 ± 0.51	15.05 ± 1.97
	SCRUB	97.21 ± 2.42	43.45 ± 12.73	93.25 ± 4.88	43.27 ± 12.23	89.36 ± 3.80	42.07 ± 10.54
	BADT	94.51 ± 0.06	50.40 ± 4.07	92.88 ± 0.15	52.83 ± 4.59	89.67 ± 0.15	50.00 ± 3.83
	SSD	95.45 ± 0.46	87.40 ± 6.10	95.49 ± 0.48	84.92 ± 7.21	90.89 ± 0.42	80.50 ± 6.42
	BS	90.36 ± 0.35	4.47 ± 0.00	90.65 ± 0.24	33.39 ± 0.75	86.24 ± 0.18	31.80 ± 1.65
	BE	87.78 ± 0.12	4.28 ± 0.66	88.02 ± 0.15	6.55 ± 0.67	83.78 ± 0.15	8.27 ± 1.32
	CPU	99.99 ± 0.00	77.84 ± 0.74	91.36 ± 0.03	76.75 ± 0.48	88.46 ± 0.13	72.36 ± 0.41
	EFACE	93.73 ± 0.14	1.37 ± 0.09	89.60 ± 0.15	5.61 ± 0.35	85.64 ± 0.13	6.63 ± 0.39
Out	$\nabla\tau$	98.87 ± 0.30	72.33 ± 2.59	89.77 ± 0.40	86.38 ± 0.80	87.33 ± 0.30	81.96 ± 0.76
	AMN	100.00 ± 0.00	0.00 ± 0.00	90.43 ± 0.29	47.52 ± 2.75	87.51 ± 0.40	47.65 ± 1.75
	SalUn	100.00 ± 0.00	0.31 ± 0.70	81.77 ± 0.38	15.25 ± 1.82	79.95 ± 0.44	15.21 ± 2.21
	SCRUB	99.82 ± 0.08	0.31 ± 0.70	91.84 ± 0.15	12.31 ± 0.36	88.03 ± 0.25	11.61 ± 0.58
	BADT	87.45 ± 0.21	65.41 ± 2.09	94.68 ± 0.05	82.23 ± 0.55	90.39 ± 0.09	77.51 ± 0.55
	SSD	87.33 ± 0.00	75.47 ± 0.00	96.16 ± 0.00	93.32 ± 0.00	91.53 ± 0.01	88.19 ± 0.11
	BS	78.99 ± 0.00	3.77 ± 0.00	88.85 ± 0.00	19.99 ± 0.01	85.18 ± 0.02	21.07 ± 0.00
	BE	79.44 ± 0.28	0.00 ± 0.00	87.62 ± 0.30	18.45 ± 0.35	83.84 ± 0.25	16.55 ± 1.02
	CPU	100.00 ± 0.00	72.64 ± 1.81	85.41 ± 0.05	74.09 ± 0.23	83.54 ± 0.10	73.04 ± 0.43
	EFACE	96.90 ± 0.23	2.20 ± 1.69	89.28 ± 0.23	17.42 ± 1.50	85.63 ± 0.41	17.35 ± 1.25

Table 7: ImageNet100, RepVGG-a2 out-sample *cluster*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten clusters. MIA scores (percentage) and time efficiency (seconds) results.

Method	MIA Diff.↓	Tsec↓
$\nabla\tau$	0.06 ± 0.05	84.46 ± 1.51
AMN	0.06 ± 0.02	299.85 ± 2.51
SalUn	0.07 ± 0.05	55.47 ± 0.64
SCRUB	0.09 ± 0.04	98.18 ± 0.32
BADT	0.09 ± 0.04	41.54 ± 0.83
SSD	0.04 ± 0.02	569.70 ± 0.96
BS	0.06 ± 0.02	23.06 ± 0.53
BE	0.11 ± 0.02	22.14 ± 0.48
CPU	0.10 ± 0.02	327.27 ± 146.37
EFACE	0.05 ± 0.03	340.64 ± 1.12

I.2.4 SENSITIVITY TO c, d, α, AND $|\mathcal{D}_f|$

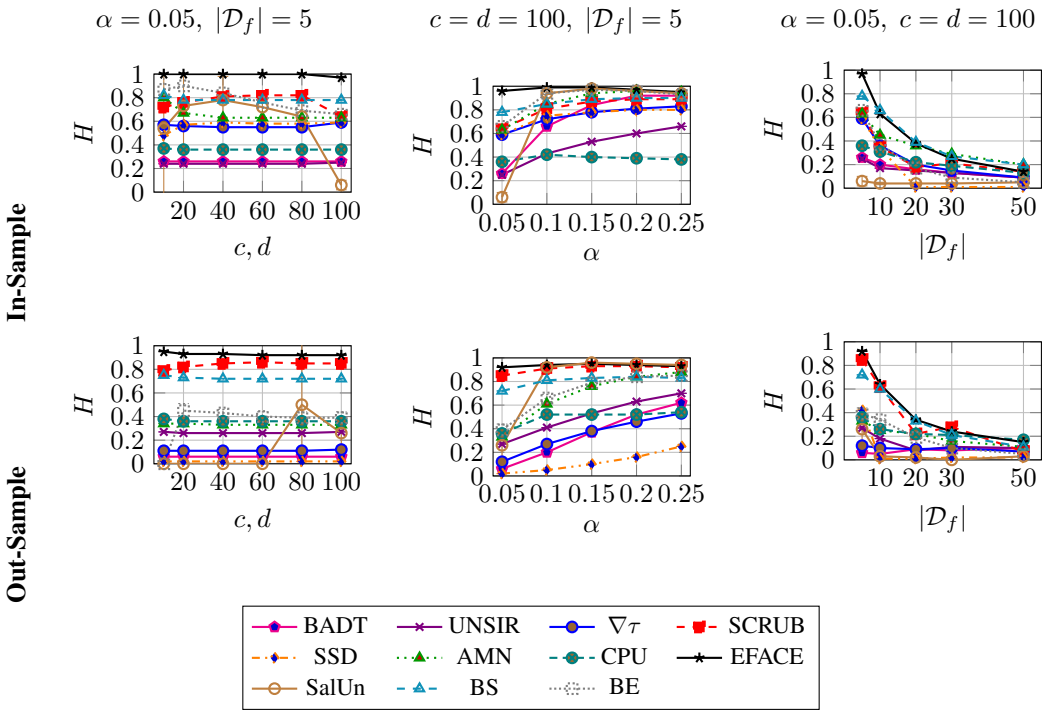


Figure 5: ImageNet100: 5 classes forgetting. (Top): In-sample results. (Bottom): Out-sample results. (Left): H vs. $c = d$. (Middle): H vs. α. (Right): H vs. $|\mathcal{D}_f|$.

In Fig. 5, we examine the effects of varying each parameter while holding others fixed: the critical set sizes c, d (left), the miscoverage tolerance α (middle), and the number of forgotten classes $|\mathcal{D}_f|$ (right). When we vary the number of forgotten classes, the size of the forgotten set increases, which relatively increases π_f. By the bound in Lemma 2, the maximum possible value of β decreases accordingly. That justifies the drop in H as the size of $|\mathcal{D}_f|$ increases, which holds for all methods whose H is not already close to zero. When α increases, the conformal predictor has more tolerance of having miscovered points while still marginally satisfying (4). Meanwhile, the larger α is, the looser the bound on β becomes by Lemma 2. Therefore, the level of miscoverage on the forget sets increases, hence H increases. All methods show the same tendency. The effect of the critical set sizes c and d is more subtle and depends on both the model and the unlearning method. When the original model performs very well—such as RepVGG-A2 on ImageNet100, which achieves 90+ accuracy on test data—most methods produce substantially small prediction sets (average size ≤ 10). In such cases, varying c and d has little effect on coverage and miscoverage, since the metrics at the full set size already capture the total coverage and miscoverage levels. EFACE produces larger prediction sets, particularly on the forget data (≈ 30), but since its H is already very high (0.95+), the metrics remain stable across different thresholds. However, this stability does not hold universally. For instance, in Table 2, setting $c = d = 50$ reveals that $\nabla\tau$ collapses in terms of coverage and miscoverage, exposing deficiencies that would remain hidden if only full set sizes were tested. This underscores the importance of thorough conformal unlearning evaluation: in practice, unlearners should test across a range of c and d values. We note that EFACE consistently outperforms other methods when α is small (e.g., 0.05), which is the regime of practical interest and the most common setting in the literature. The corresponding results for CIFAR100 and 20NewsGroups are shown in Figs. 6 and 7, and they follow the same trends.

I.2.5 CIFAR100 AND 20NEWSGROUPS

Tables 8 and 9 report cluster-wise and class-wise forgetting on CIFAR100, and Table 10 reports class-wise forgetting on 20NewsGroups, so the observations made for ImageNet100 in Section 5.1 carry over to these datasets.

Table 8: CIFAR100, RepVGG-a2 *cluster*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten clusters. Coverage/miscoverage results.

Split	Method	$\mathsf{C}_c(\mathcal{D}_r) \uparrow$	$\mathsf{M}_d(\mathcal{D}_f) \uparrow$	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$
	OR	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.06 ± 0.00	0.00 ± 0.00
In	$\nabla\tau$	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.00	0.85 ± 0.01	0.97 ± 0.00
	AMN	1.00 ± 0.00	0.97 ± 0.01	1.00 ± 0.00	0.19 ± 0.02	0.95 ± 0.00	0.17 ± 0.02	0.39 ± 0.02
	SalUn	1.00 ± 0.00	0.61 ± 0.36	1.00 ± 0.00	0.14 ± 0.03	0.95 ± 0.00	0.22 ± 0.03	0.37 ± 0.05
	SCRUB	1.00 ± 0.00	0.05 ± 0.02	0.99 ± 0.00	0.05 ± 0.01	0.95 ± 0.00	0.08 ± 0.01	0.11 ± 0.02
	BADT	1.00 ± 0.00	0.02 ± 0.02	1.00 ± 0.00	0.02 ± 0.01	0.95 ± 0.00	0.16 ± 0.01	0.04 ± 0.02
	SSD	1.00 ± 0.00	0.95 ± 0.04	1.00 ± 0.00	0.87 ± 0.02	0.97 ± 0.00	0.63 ± 0.01	0.88 ± 0.01
	BS	1.00 ± 0.00	0.66 ± 0.00	1.00 ± 0.00	0.13 ± 0.00	0.95 ± 0.00	0.23 ± 0.00	0.36 ± 0.00
	BE	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.09 ± 0.00	0.00 ± 0.00
	CPU	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.03 ± 0.01	0.00 ± 0.00
	EFACE	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.97 ± 0.00	0.86 ± 0.01	0.97 ± 0.00
Out	$\nabla\tau$	1.00 ± 0.00	0.24 ± 0.16	1.00 ± 0.00	0.03 ± 0.06	0.95 ± 0.00	0.08 ± 0.05	0.11 ± 0.14
	AMN	1.00 ± 0.00	0.92 ± 0.03	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.04 ± 0.01	0.02 ± 0.01
	SalUn	1.00 ± 0.00	0.11 ± 0.04	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.03 ± 0.01	0.02 ± 0.01
	SCRUB	1.00 ± 0.00	0.12 ± 0.05	0.96 ± 0.01	0.14 ± 0.05	0.94 ± 0.01	0.14 ± 0.07	0.23 ± 0.05
	BADT	0.94 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.01 ± 0.00	0.00 ± 0.00
	SSD	0.96 ± 0.00	0.45 ± 0.02	1.00 ± 0.00	0.36 ± 0.02	0.96 ± 0.00	0.43 ± 0.03	0.58 ± 0.01
	BS	0.95 ± 0.00	0.85 ± 0.00	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.04 ± 0.00	0.05 ± 0.00
	BE	0.94 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.08 ± 0.00	0.00 ± 0.00
	CPU	1.00 ± 0.00	0.02 ± 0.01	1.00 ± 0.00	0.01 ± 0.00	0.96 ± 0.00	0.04 ± 0.01	0.03 ± 0.00
	EFACE	0.99 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.82 ± 0.06	0.97 ± 0.00	0.78 ± 0.03	0.92 ± 0.02

Table 9: CIFAR100, RepVGG-a2 *class*-wise forgetting with $c = d = 50$, $\alpha = 0.05$, and 5 forgotten classes. Coverage/miscoverage results.

Split	Method	$\mathsf{C}_c(\mathcal{D}_r) \uparrow$	$\mathsf{M}_d(\mathcal{D}_f) \uparrow$	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$
	OR	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.05 ± 0.00	0.00 ± 0.00
In	$\nabla\tau$	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00	1.00 ± 0.00	1.00 ± 0.00
	AMN	1.00 ± 0.00	0.97 ± 0.01	1.00 ± 0.00	0.19 ± 0.03	0.96 ± 0.00	0.32 ± 0.04	0.48 ± 0.03
	SalUn	1.00 ± 0.00	0.44 ± 0.19	1.00 ± 0.00	0.06 ± 0.01	0.96 ± 0.00	0.36 ± 0.04	0.24 ± 0.04
	SCRUB	1.00 ± 0.00	0.04 ± 0.04	0.99 ± 0.01	0.05 ± 0.04	0.95 ± 0.00	0.15 ± 0.03	0.12 ± 0.05
	BADT	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.28 ± 0.01	0.00 ± 0.00
	SSD	1.00 ± 0.00	0.93 ± 0.03	1.00 ± 0.00	0.90 ± 0.01	0.97 ± 0.00	0.94 ± 0.01	0.96 ± 0.01
	UNSIR	1.00 ± 0.00	0.01 ± 0.00	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.06 ± 0.01	0.02 ± 0.01
	BS	1.00 ± 0.00	0.69 ± 0.00	1.00 ± 0.00	0.12 ± 0.00	0.95 ± 0.00	0.29 ± 0.00	0.37 ± 0.00
	BE	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.23 ± 0.00	0.00 ± 0.00
	CPU	1.00 ± 0.00	0.06 ± 0.02	1.00 ± 0.00	0.05 ± 0.02	0.96 ± 0.01	0.20 ± 0.02	0.13 ± 0.03
	EFACE	1.00 ± 0.00	0.98 ± 0.01	1.00 ± 0.00	0.89 ± 0.02	0.98 ± 0.00	0.92 ± 0.01	0.96 ± 0.00
Out	$\nabla\tau$	1.00 ± 0.00	0.40 ± 0.24	1.00 ± 0.00	0.09 ± 0.11	0.96 ± 0.00	0.36 ± 0.15	0.31 ± 0.21
	AMN	1.00 ± 0.00	0.95 ± 0.02	1.00 ± 0.00	0.04 ± 0.02	0.96 ± 0.00	0.27 ± 0.02	0.19 ± 0.05
	SalUn	1.00 ± 0.00	0.32 ± 0.18	1.00 ± 0.00	0.07 ± 0.01	0.96 ± 0.00	0.26 ± 0.01	0.25 ± 0.02
	SCRUB	1.00 ± 0.00	0.33 ± 0.03	1.00 ± 0.00	0.05 ± 0.02	0.96 ± 0.00	0.16 ± 0.01	0.18 ± 0.04
	BADT	0.95 ± 0.00	0.04 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.10 ± 0.00	0.00 ± 0.00
	SSD	0.97 ± 0.00	0.46 ± 0.02	1.00 ± 0.00	0.36 ± 0.00	0.96 ± 0.00	0.45 ± 0.01	0.59 ± 0.01
	UNSIR	1.00 ± 0.00	0.02 ± 0.02	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.04 ± 0.01	0.04 ± 0.01
	BS	0.95 ± 0.00	0.90 ± 0.00	1.00 ± 0.00	0.06 ± 0.00	0.95 ± 0.00	0.23 ± 0.00	0.24 ± 0.00
	BE	0.95 ± 0.00	0.09 ± 0.01	1.00 ± 0.00	0.01 ± 0.00	0.95 ± 0.00	0.17 ± 0.00	0.05 ± 0.00
	CPU	1.00 ± 0.00	0.27 ± 0.02	1.00 ± 0.00	0.11 ± 0.01	0.97 ± 0.00	0.24 ± 0.02	0.31 ± 0.01
	EFACE	0.99 ± 0.00	0.98 ± 0.01	1.00 ± 0.00	0.68 ± 0.04	0.97 ± 0.00	0.79 ± 0.03	0.88 ± 0.01

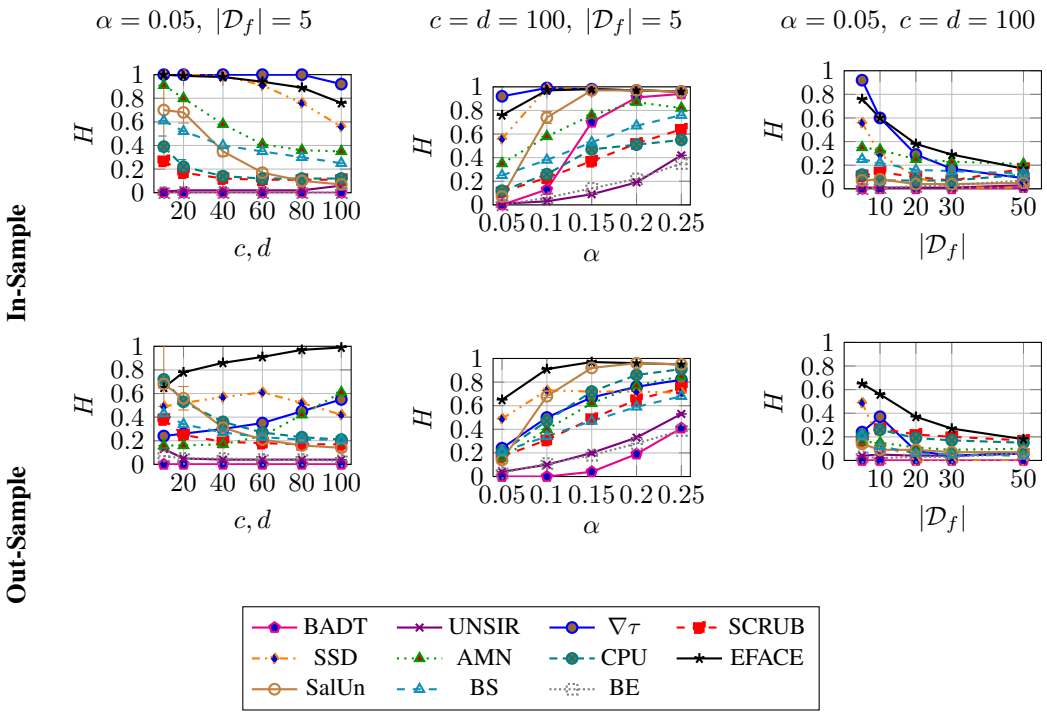


Figure 6: CIFAR100, RepVGG-a2: 5 classes forgetting. (Top): In-sample results. (Bottom): Out-sample results. (Left): H vs. $c = d$. (Middle): H vs. α. (Right): H vs. $|\mathcal{D}_f|$.

Table 10: 20NewsGroups *class*-wise forgetting with $c = d = 20$, $\alpha = 0.05$, and 1 forgotten class. Coverage/miscoverage results.

Split	Method	$\mathsf{C}_c(\mathcal{D}_r) \uparrow$	$\mathsf{M}_d(\mathcal{D}_f) \uparrow$	$\mathsf{C}_c(\mathcal{T}_r) \uparrow$	$\mathsf{M}_d(\mathcal{T}_f) \uparrow$	$\mathsf{C}_c(\mathcal{V}_r) \uparrow$	$\mathsf{M}_d(\mathcal{V}_f) \uparrow$	$H \uparrow$
	OR	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.05 ± 0.00	0.00 ± 0.00
In	$\nabla\tau$	1.00 ± 0.00	0.94 ± 0.01	1.00 ± 0.00	0.92 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.00
	AMN	1.00 ± 0.00	0.74 ± 0.02	1.00 ± 0.00	0.16 ± 0.03	0.97 ± 0.00	0.23 ± 0.09	0.40 ± 0.05
	SalUn	1.00 ± 0.00	0.03 ± 0.03	1.00 ± 0.00	0.13 ± 0.02	0.98 ± 0.00	0.18 ± 0.02	0.13 ± 0.08
	SCRUB	1.00 ± 0.00	0.94 ± 0.05	1.00 ± 0.00	0.91 ± 0.05	1.00 ± 0.00	0.95 ± 0.06	0.96 ± 0.02
	BADT	1.00 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.97 ± 0.00	0.07 ± 0.02	0.00 ± 0.00
	SSD	0.96 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.00 ± 0.00	0.00 ± 0.00
	BS	1.00 ± 0.00	0.91 ± 0.01	1.00 ± 0.00	0.74 ± 0.01	0.99 ± 0.00	0.84 ± 0.00	0.90 ± 0.00
	BE	1.00 ± 0.00	0.01 ± 0.01	1.00 ± 0.00	0.00 ± 0.00	0.95 ± 0.00	0.11 ± 0.00	0.01 ± 0.01
	CPU	1.00 ± 0.00	0.01 ± 0.01	1.00 ± 0.00	0.02 ± 0.02	0.97 ± 0.00	0.16 ± 0.00	0.03 ± 0.03
	EFACE	1.00 ± 0.00	0.97 ± 0.01	1.00 ± 0.00	0.95 ± 0.01	1.00 ± 0.00	1.00 ± 0.00	0.99 ± 0.00
Out	$\nabla\tau$	1.00 ± 0.00	0.95 ± 0.00	1.00 ± 0.00	0.96 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.00
	AMN	1.00 ± 0.00	0.84 ± 0.06	1.00 ± 0.00	0.15 ± 0.02	0.98 ± 0.00	0.33 ± 0.09	0.44 ± 0.04
	SalUn	1.00 ± 0.00	0.03 ± 0.02	0.98 ± 0.00	0.18 ± 0.03	0.97 ± 0.00	0.26 ± 0.04	0.14 ± 0.07
	SCRUB	1.00 ± 0.00	0.82 ± 0.08	1.00 ± 0.00	0.82 ± 0.08	0.99 ± 0.00	0.88 ± 0.06	0.91 ± 0.03
	BADT	0.94 ± 0.00	0.00 ± 0.00	1.00 ± 0.00	0.00 ± 0.00	0.97 ± 0.00	0.05 ± 0.00	0.00 ± 0.00
	SSD	0.95 ± 0.00	0.01 ± 0.01	1.00 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.05 ± 0.00	0.00 ± 0.00
	BS	0.98 ± 0.00	0.83 ± 0.01	1.00 ± 0.00	0.32 ± 0.03	0.99 ± 0.00	0.75 ± 0.02	0.69 ± 0.02
	BE	0.96 ± 0.00	0.16 ± 0.04	1.00 ± 0.00	0.00 ± 0.00	0.96 ± 0.00	0.11 ± 0.00	0.02 ± 0.01
	CPU	1.00 ± 0.00	0.15 ± 0.04	1.00 ± 0.00	0.03 ± 0.01	0.96 ± 0.01	0.26 ± 0.04	0.13 ± 0.02
	EFACE	1.00 ± 0.00	0.95 ± 0.00	1.00 ± 0.00	0.96 ± 0.00	1.00 ± 0.00	1.00 ± 0.00	0.98 ± 0.00

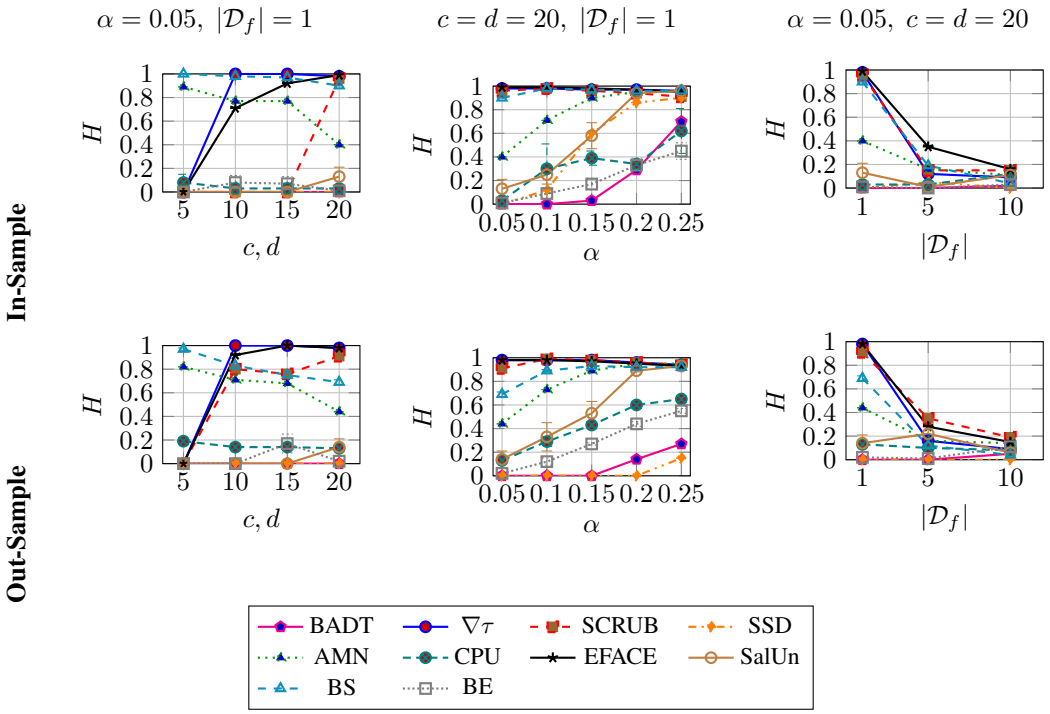


Figure 7: 20NewsGroups: 1 class forgetting. (Top): In-sample results. (Bottom): Out-sample results. (Left): H vs. $c = d$. (Middle): H vs. α. (Right): H vs. $|\mathcal{D}_f|$.

I.2.6 Comparison with RT and PABI

Tables 11 to 13 extend the comparison of Table 3 to ImageNet100 class-wise forgetting and to CIFAR100 cluster-wise and class-wise forgetting, and they use the ResNet18 models trained as described in Appendix I.1.

Table 11: ImageNet100, ResNet18 *class*-wise forgetting with $c = d = 100$, $\alpha = 0.05$, and 5 forgotten classes. Coverage/miscoverage, accuracy (%) and unlearning time (seconds) results.

Split	Method	$\mathsf{C}_c(\mathcal{T}_r)\uparrow$	$\mathsf{M}_d(\mathcal{T}_f)\uparrow$	$\mathsf{C}_c(\mathcal{V}_r)\uparrow$	$\mathsf{M}_d(\mathcal{V}_f)\uparrow$	$H\uparrow$	$A(\mathcal{T}_r)\uparrow$	$A(\mathcal{T}_f)\downarrow$	$A(\mathcal{V}_r)\uparrow$	$A(\mathcal{V}_f)\downarrow$	Tsec↓
	RT	1.00 ± 0.00	0.55 ± 0.01	0.98 ± 0.00	0.53 ± 0.02	0.63 ± 0.03	99.57 ± 0.00	0.00 ± 0.00	72.14 ± 0.51	0.00 ± 0.00	18172.47 ± 221.71
	PABI	1.00 ± 0.00	0.57 ± 0.02	0.98 ± 0.00	0.58 ± 0.02	0.67 ± 0.04	99.58 ± 0.00	0.00 ± 0.00	68.60 ± 0.32	0.00 ± 0.00	9182.29 ± 48.87
In	EFACE	1.00 ± 0.00	0.34 ± 0.01	0.98 ± 0.00	0.52 ± 0.01	0.65 ± 0.01	99.55 ± 0.00	0.00 ± 0.00	68.19 ± 0.01	0.00 ± 0.00	2634.28 ± 20.15
Out	EFACE	1.00 ± 0.00	0.23 ± 0.03	0.97 ± 0.00	0.38 ± 0.02	0.54 ± 0.03	96.67 ± 0.48	3.80 ± 0.98	62.86 ± 0.26	0.30 ± 0.25	330.78 ± 3.02

Table 12: CIFAR100, ResNet18 *cluster*-wise forgetting with $c = d = 100$, $\alpha = 0.05$, and 5 forgotten clusters. Coverage/miscoverage, accuracy (%) and unlearning time (seconds) results.

Split	Method	$\mathsf{C}_c(\mathcal{T}_r)\uparrow$	$\mathsf{M}_d(\mathcal{T}_f)\uparrow$	$\mathsf{C}_c(\mathcal{V}_r)\uparrow$	$\mathsf{M}_d(\mathcal{V}_f)\uparrow$	$H\uparrow$	$A(\mathcal{T}_r)\uparrow$	$A(\mathcal{T}_f)\downarrow$	$A(\mathcal{V}_r)\uparrow$	$A(\mathcal{V}_f)\downarrow$	Tsec↓
	RT	1.00 ± 0.00	0.04 ± 0.00	0.95 ± 0.00	0.05 ± 0.01	0.09 ± 0.01	99.67 ± 0.01	59.27 ± 0.46	54.19 ± 0.28	57.64 ± 0.49	560.12 ± 8.70
	PABI	1.00 ± 0.00	0.05 ± 0.00	0.94 ± 0.00	0.05 ± 0.01	0.08 ± 0.01	99.99 ± 0.00	51.15 ± 0.79	52.46 ± 0.43	55.56 ± 0.65	558.52 ± 16.00
In	EFACE	1.00 ± 0.00	0.19 ± 0.00	0.95 ± 0.00	0.19 ± 0.02	0.40 ± 0.01	99.12 ± 0.14	78.03 ± 0.64	47.19 ± 0.75	38.85 ± 2.45	101.20 ± 1.12
Out	EFACE	1.00 ± 0.00	0.13 ± 0.00	0.95 ± 0.00	0.18 ± 0.02	0.35 ± 0.01	98.58 ± 0.21	82.53 ± 0.33	49.29 ± 0.44	40.81 ± 2.07	76.15 ± 1.39

I.3 Limitations of EFACE

We note the following limitations of EFACE.

- Since EFACE relies on solving an optimization problem over the unlearning set $\mathcal{D}_{\text{unlearn}}$, its generalization cannot be guaranteed for data outside the unlearning set. When the unlearning is in-sample, we notice that the performance of EFACE increases as it is optimized over a more representative dataset.

Table 13: CIFAR100, ResNet18 *class*-wise forgetting with $c = d = 100$, $\alpha = 0.05$, and 5 forgotten classes. Coverage/miscoverage, accuracy (%) and unlearning time (seconds) results.

Split	Method	$\mathsf{C}_c(\mathcal{T}_r)\uparrow$	$\mathsf{M}_d(\mathcal{T}_f)\uparrow$	$\mathsf{C}_c(\mathcal{V}_r)\uparrow$	$\mathsf{M}_d(\mathcal{V}_f)\uparrow$	$H\uparrow$	$A(\mathcal{T}_r)\uparrow$	$A(\mathcal{T}_f)\downarrow$	$A(\mathcal{V}_r)\uparrow$	$A(\mathcal{V}_f)\downarrow$	Tsec↓
	RT	1.00 ± 0.00	0.80 ± 0.01	0.99 ± 0.00	0.84 ± 0.01	0.89 ± 0.01	99.99 ± 0.00	0.00 ± 0.00	56.67 ± 0.11	0.00 ± 0.00	594.72 ± 13.50
	PABI	1.00 ± 0.00	0.72 ± 0.04	0.98 ± 0.00	0.72 ± 0.05	0.83 ± 0.02	99.99 ± 0.00	0.00 ± 0.00	54.62 ± 0.99	0.00 ± 0.00	550.22 ± 1.93
In	EFACE	1.00 ± 0.00	0.50 ± 0.03	0.98 ± 0.00	0.62 ± 0.03	0.78 ± 0.01	99.76 ± 0.07	0.07 ± 0.06	51.52 ± 0.28	0.00 ± 0.00	28.46 ± 0.30
Out	EFACE	1.00 ± 0.00	0.29 ± 0.02	0.97 ± 0.00	0.51 ± 0.02	0.63 ± 0.02	96.21 ± 0.60	1.06 ± 0.45	51.23 ± 0.47	0.00 ± 0.00	21.76 ± 0.16

- EFACE uses a specific conformity scoring function for unlearning. In our experiments we used the softmax probability of the label as a scoring function. As described in the discussion under Appendix E, if the downstream decision maker uses a conformity scoring function that is different in a strange manner from the one used in EFACE, then EFACE cannot be guaranteed to induce proper conformal unlearning.

J Further Sensitivity Analysis

Remark J.1 (Scope of the set-size condition)**.** *The objective that EFACE solves,* (15)*, is unconstrained, so the set-levels* c *and* d *play no role during unlearning and enter only through the CF and mCF metrics of* (9) *and* (10) *used for evaluation. All results in the main text and in Appendix I.2 are therefore obtained with no set-size condition, which corresponds to the setting* $\rho = 0$ *below. In an earlier formulation we also constrained the average prediction-set size during unlearning, and we retain that study here as an* ablation*: the panels involving* ρ *and* c *in Figs. 8 to 11 add a penalty term* $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$ *to the objective, where* $\bar{s}$ *is a differentiable surrogate for the average set size. The ablation shows what such a condition would buy and what it would cost, and it is reported for completeness only; it is not part of the method evaluated elsewhere in the paper.*

We present a sensitivity analysis in the following plots. We analyze the sensitivity of EFACE to various hyperparameters in terms of retained-data coverage $\mathsf{C}_{\mathcal{D}_r}$ and $\mathsf{C}_{\mathcal{V}_r}$ and accuracy $A_{\mathcal{D}_r}$ and $A_{\mathcal{V}_r}$, the forgotten-data miscoverage $\mathsf{M}_{\mathcal{D}_f}$ and $\mathsf{M}_{\mathcal{V}_f}$ and accuracy $A_{\mathcal{D}_f}$ and $A_{\mathcal{V}_f}$, and the harmonic mean H. Experiments were conducted using 3 random seeds on CIFAR100 class-wise forgetting (Fig. 8), CIFAR100 cluster-wise forgetting (Fig. 9), ImageNet100 class-wise forgetting (Fig. 10) and ImageNet100 cluster-wise forgetting (Fig. 11), forgetting 5 classes or clusters in every case. We also report in Fig. 12 the average prediction-set sizes of the subsets $\mathcal{D}_r$, $\mathcal{D}_f$, $\mathcal{V}_r$ and $\mathcal{V}_f$ before and after unlearning on the same four settings.

In Fig. 8 top left, we vary the condition penalty term ρ of the set-size ablation of Remark J.1, which weights the set-size condition in the augmented Lagrangian. Two remarks are in order. First, ρ and the set-size condition $[c, d]$ are not separable: the condition reaches the objective only through the coefficient $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$, so with $\rho = 0$ the condition is inactive for every $[c, d]$, and with a cap that is never violated the value of ρ is equally immaterial. Sweeping either one alone therefore produces a flat curve that says nothing about the method. Each panel here consequently fixes the other member of the pair, as stated in the caption. Second, once the pair is active, ρ is influential: increasing it from 10^{-4} to 1 drives H from 0.69 to 0.00 as the escalating penalty overwhelms the coverage objectives and degrades the model. Small values of ρ are safe. Generally, values above 10^{-2} place a huge weight on the efficiency condition and therefore are considered too strong to leave room for effective optimization, an effect that is expected.

In Fig. 8 top right, we vary the unlearned-model parameter regularization constant γ. This term controls the penalty on the deviations of the parameters θ_u from the original parameters θ_o. We cannot find a linear correlation between the metrics and the value of γ. However, we notice that choosing γ close to 1 causes a clear drop in the performance of the model by decreasing the level of miscoverage frequency on the forgotten data. That is expected as choosing a large γ value lowers the flexibility of EFACE in updating the unlearned model's parameters away from the original parameters, which makes the unlearning process less effective in achieving the desired coverage/miscoverage objectives. However, large γ helps stabilize the unlearning process which results in higher accuracy on the validation subsets (less utility drop of the model).

In Fig. 8 bottom left, we vary the steepness of the surrogate indicator. Recall that the ratio of covered/miscovered points is found using the indicator function on whether their true label is

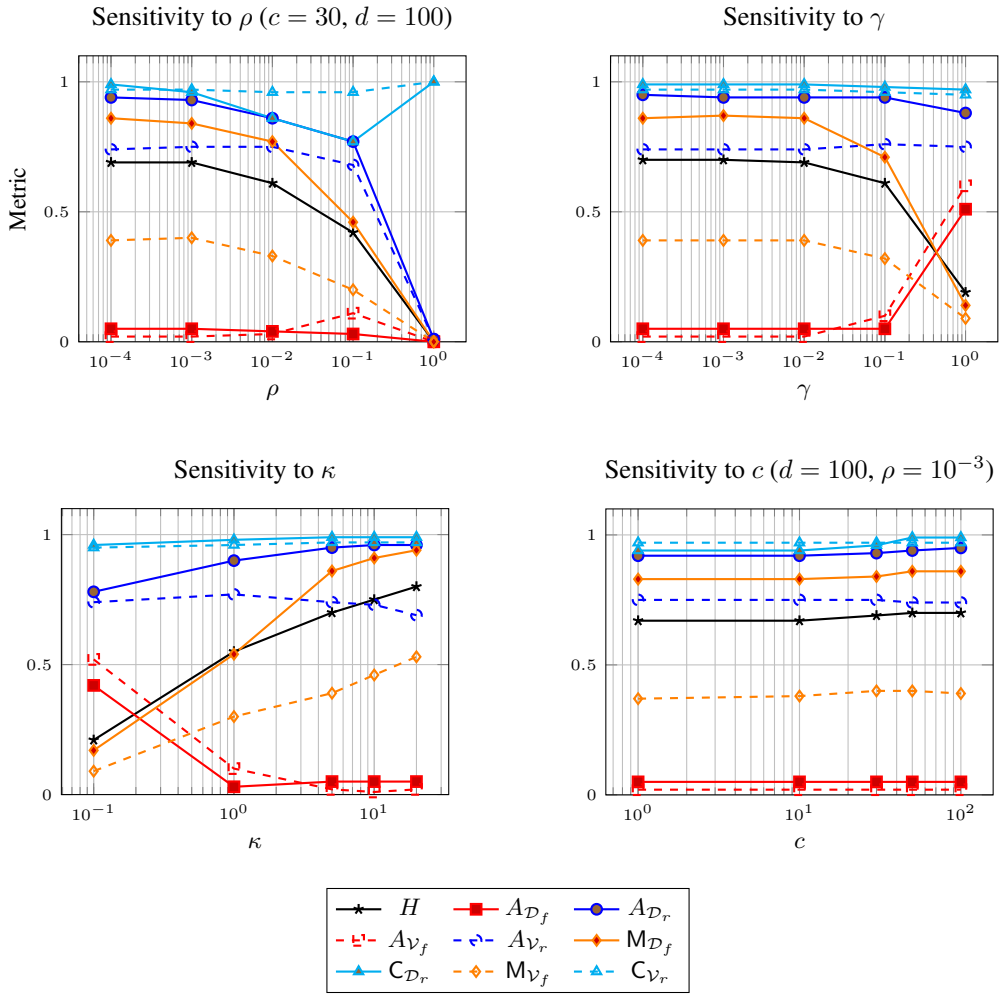


Figure 8: Sensitivity analysis of EFACE hyperparameters on CIFAR100, RepVGG-a2, class-wise forgetting of 5 classes. We vary one parameter at a time while keeping the others fixed, and average over 3 seeds. Top row: ρ and γ. Bottom row: κ and c. Because the set-size term enters the objective as $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$, neither ρ nor c can act alone: the ρ panel therefore fixes the binding cap $c = 30$ ($d = 100$), and the c panel fixes $\rho = 10^{-3}$.

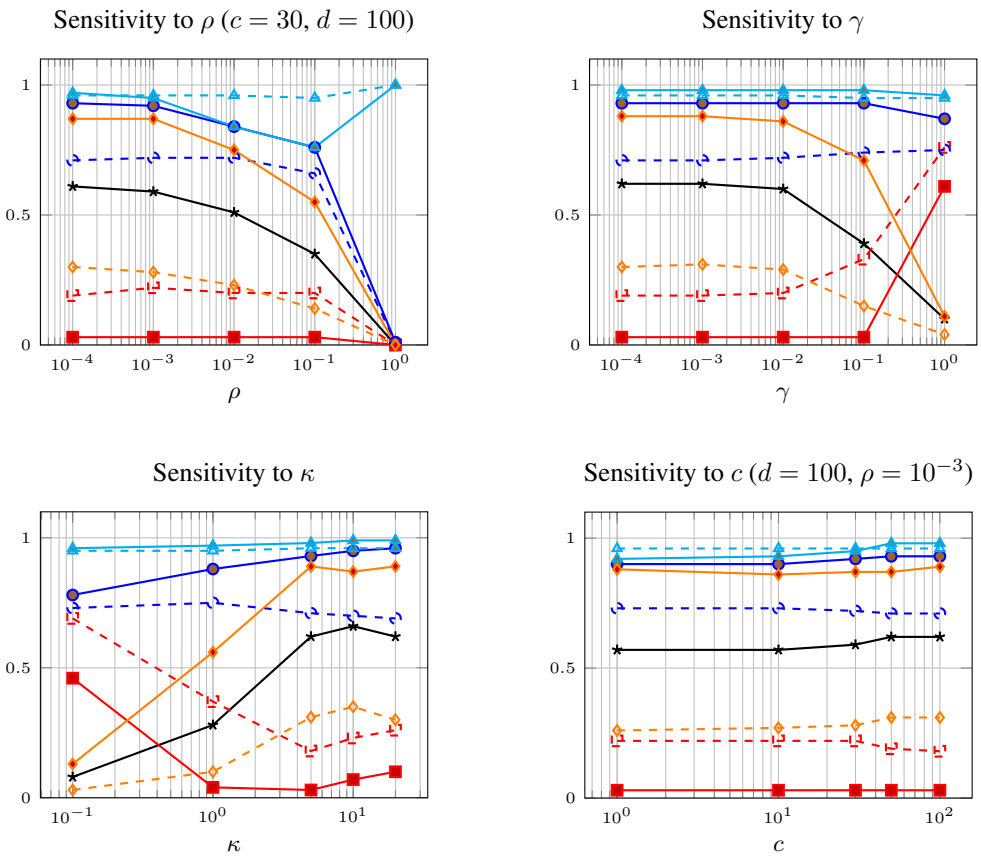


Figure 9: Sensitivity analysis of EFACE hyperparameters on CIFAR100, RepVGG-a2, cluster-wise forgetting of 5 clusters. We vary one parameter at a time while keeping the others fixed, and average over 3 seeds. Top row: ρ and γ. Bottom row: κ and c. Because the set-size term enters the objective as $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$, neither ρ nor c can act alone: the ρ panel therefore fixes the binding cap $c = 30$ ($d = 100$), and the c panel fixes $\rho = 10^{-3}$. The legend is shared with Fig. 8.

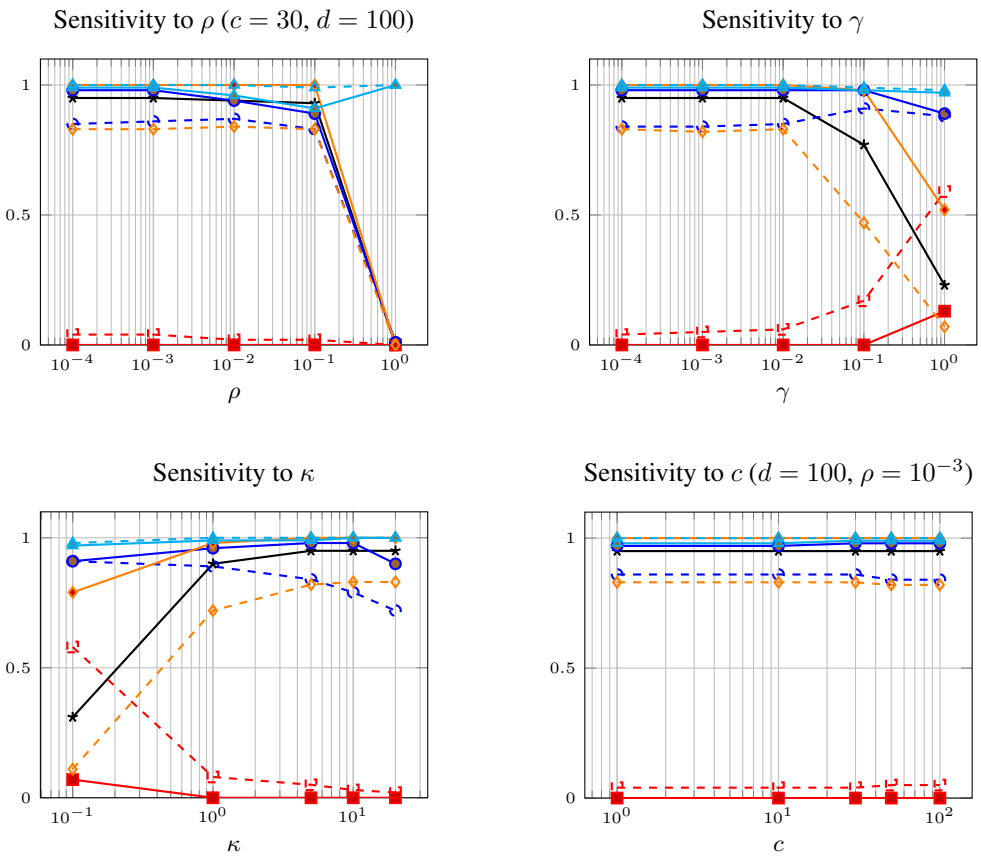


Figure 10: Sensitivity analysis of EFACE hyperparameters on ImageNet100, RepVGG-a2, class-wise forgetting of 5 classes. We vary one parameter at a time while keeping the others fixed, and average over 3 seeds. Top row: ρ and γ. Bottom row: κ and c. Because the set-size term enters the objective as $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$, neither ρ nor c can act alone: the ρ panel therefore fixes the binding cap $c = 30$ ($d = 100$), and the c panel fixes $\rho = 10^{-3}$. The legend is shared with Fig. 8.

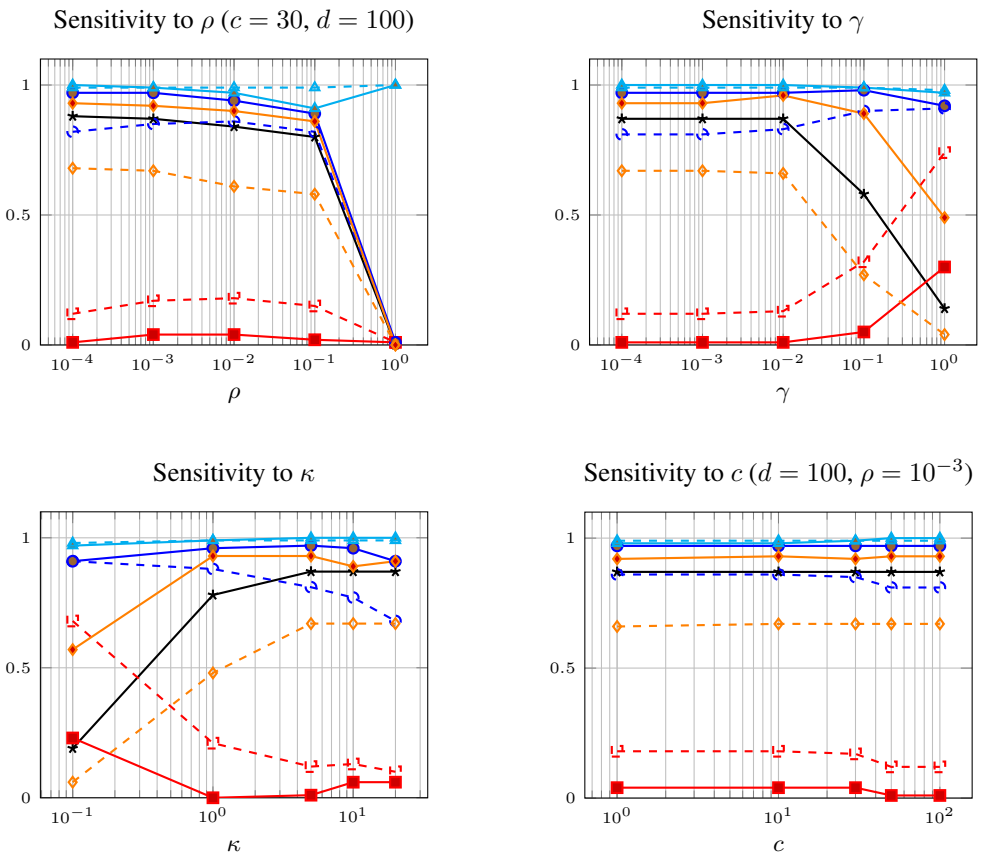


Figure 11: Sensitivity analysis of EFACE hyperparameters on ImageNet100, RepVGG-a2, cluster-wise forgetting of 5 clusters. We vary one parameter at a time while keeping the others fixed, and average over 3 seeds. Top row: ρ and γ. Bottom row: κ and c. Because the set-size term enters the objective as $(\lambda + \rho\,[\,\bar{s} - c\,]_+)$, neither ρ nor c can act alone: the ρ panel therefore fixes the binding cap $c = 30$ ($d = 100$), and the c panel fixes $\rho = 10^{-3}$. The legend is shared with Fig. 8.

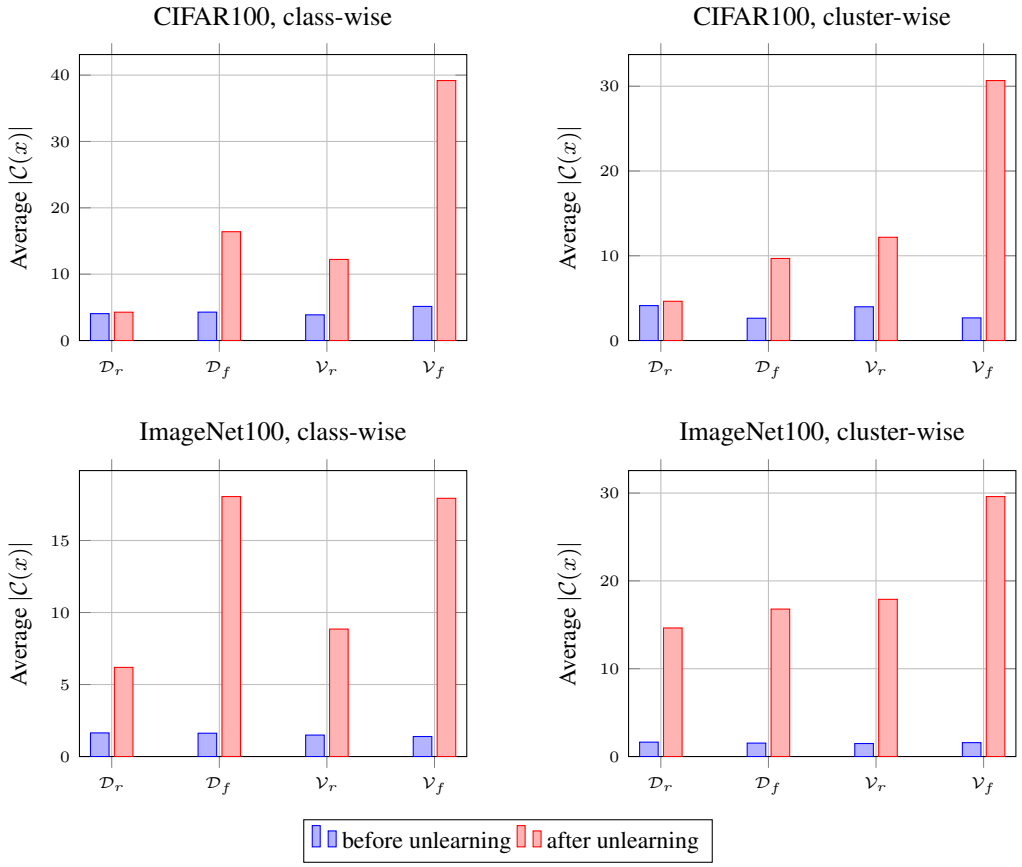


Figure 12: Average conformal prediction-set size on the retained and forgotten subsets before and after unlearning with EFACE (RepVGG-a2, forgetting 5 classes/clusters, 3 seeds, $\alpha = 0.05$). Each model is calibrated with its own quantile on the same calibration split. Before unlearning the four subsets are indistinguishable; after unlearning the sets grow sharply on the forgotten subsets $\mathcal{D}_f$ and $\mathcal{V}_f$ while staying essentially unchanged on the retained subset $\mathcal{D}_r$, which is precisely the behavior the conformal unlearning objective asks for. The 'before' bars carry no seed spread because the original model is the same for all seeds.

included in or excluded from the prediction set. The indicator function is a step function that is non-differentiable. Hence, we use the sigmoid function with steepness κ to approximate it. We notice that generally $\kappa = 5$ is a good compromise. Decreasing κ below 5 makes the surrogate too smooth to capture the decision boundaries created by the indicator function, while increasing κ further sharpens the surrogate at the cost of smooth gradients for learning.

The last subfigure at the bottom right of each sensitivity figure shows the effect of the set-size condition c (with $d = 100$) at a fixed $\rho = 10^{-3}$. The condition is effective, and it acts exactly where it is applied: tightening c from 100 to 1 steadily shrinks the average prediction set on the retained data (CIFAR100 class-wise: 4.27 $\rightarrow$ 1.50; CIFAR100 cluster-wise: 4.63 $\rightarrow$ 1.43; ImageNet100 class-wise: 6.19 $\rightarrow$ 2.14; ImageNet100 cluster-wise: 14.64 $\rightarrow$ 2.95), while the forgotten sets, whose cap $d = 100$ is left loose, are essentially untouched. On ImageNet100 this efficiency is obtained at no measurable cost in H, whereas on CIFAR100 it costs a few points of retained coverage $\mathsf{C}_{\mathcal{D}_r}$.

The condition does, however, only act below a threshold that is a property of the surrogate rather than of the data, which is why a coarse grid can make it look irrelevant. What the constraint measures is $\bar{s} = \mathbb{E}\big[\sum_k \sigma(\kappa(p_k(x) - (1-t)))\big]$, evaluated at the epoch-wise threshold t. In all our settings $t \rightarrow 1$, so $1 - t \approx 0$ and each of the near-zero class probabilities contributes approximately $\sigma(0) = 1/2$; the surrogate is consequently pinned near $|\mathcal{Y}|/2$ regardless of the true prediction-set size. Caps above that value are never violated and leave the solution bitwise unchanged, while caps below it are never satisfiable and act as a steadily growing pressure on the set sizes. A useful rule of thumb is therefore to place the grid for c relative to $|\mathcal{Y}|/2$ rather than on an absolute scale, and never to sweep c and ρ independently.

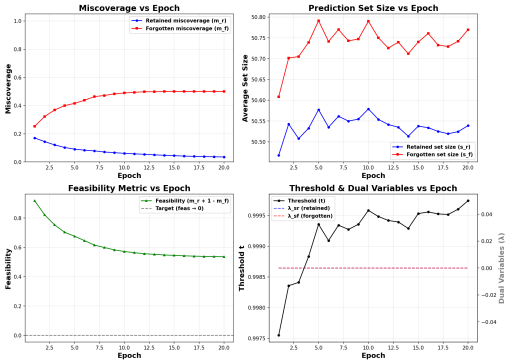
Finally, Fig. 12 isolates the effect of unlearning on the prediction sets themselves. Before unlearning, the original model produces sets of comparable size on all four subsets ($\mathcal{D}_r$: 4.05, $\mathcal{D}_f$: 4.28, $\mathcal{V}_r$: 3.87, $\mathcal{V}_f$: 5.14 on CIFAR100 class-wise), as it has no reason to treat them differently. After unlearning, the sets on the forgotten subsets grow by a large factor ($\mathcal{D}_f$: 16.40, $\mathcal{V}_f$: 39.18), reflecting the uncertainty EFACE deliberately injects on the forgotten data, whereas the retained subset is almost unaffected ($\mathcal{D}_r$: 4.27) or remains actionable ($\mathcal{V}_r$: 12.22). The gap between $\mathcal{D}_r$ and $\mathcal{V}_r$ after unlearning reflects the usual in-sample/out-of-sample difference of the retained data rather than an effect of the condition.

J.1 Empirical Convergence and Feasibility

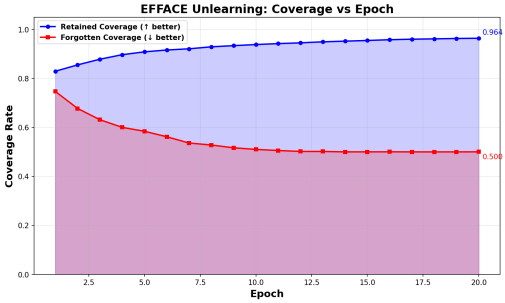
To better understand whether EFACE shows some convergence, we next show how EFACE's loss behaves against the number of epochs. The results are of ImageNet100 forgetting 10 labels. Notice in Fig. 13a how the forgotten miscoverage follows a fast increase at the beginning up to 7 epochs while the retained miscoverage decreases by the same rate. After that, both of miscoverage levels seem to converge to their optimal values. The miscoverage level over the retained set α and that over the forgotten set β are related by the probability of the forgotten data. Here, since we are forgetting 10 labels out of 100, we can assume that the probability of getting a new point belonging to the forgotten labels is ≈ 0.1. Hence, the forgotten data miscoverage level can be at most 10 times that of the retained data. Since $\alpha = 0.05$ in this case, we expect to see what is demonstrated in the figures with forget-data miscoverage 0.5 and retained-data coverage 0.05. The same convergence is shown by the feasibility plot, which tracks the objective of (15) shifted by one, that is, the miscoverage level over the retained data minus that over the forgotten data plus 1, and not the feasibility of any constraint, since (15) is unconstrained. We target 0 feasibility when β can go up to 1 (in this case it is bounded by 0.5), meaning full miscoverage over the forgotten data and no miscoverage over the retained data. Fig. 13b shows the same tendency but we replace miscoverage with coverage (flipping the plot upside down for easier reference).

J.2 Approximate Memory Requirement

EFACE requires approximately 4,116 MB of GPU memory when unlearning 10 classes from CIFAR100 on ResNet18, compared to 5,282 MB for $\nabla\tau$, 4,304 MB for SCRUB, 3,054 MB for AMN, 4,024 MB for UNSIR, and 4,032 MB for BADT. This suggests that EFACE seems not to use more memory compared to the average memory usage in the considered baselines.



(a) Plots of EFACE objective convergence.



(b) Plot of $1 - \hat{m}_r$ and $1 - \hat{m}_f$ vs epochs.

Figure 13: EFACE empirical convergence and feasibility on ImageNet100 forgetting 10 labels.

J.3 How Can c and d Be Chosen in Practice? An Illustrative Example

The following example illustrates how the set-levels c and d of the CF and mCF metrics in (9) and (10) can be chosen. Consider a 20-class document classification task. If the model produces a prediction set of size 10 that includes a confidential label intended for forgetting, the set may be deemed too large to represent a significant information leak. Conversely, a smaller prediction set of size 5 containing the same label would be more informative and thus indicative of incomplete forgetting. For retained labels, a prediction set of size 7 may still provide sufficient specificity to be practically useful. Practitioners may therefore select $d = 5$ for forgotten data and $c = 7$ for retained data, reflecting their tolerance for uncertainty in each case. Evaluating multiple (c, d) pairs can provide insights into the trade-offs between coverage and miscoverage.