Ascent Fails to Forget

Ioannis Mavrothalassitis [*1] **Pol Puigdemont** [*1] **Noam Levi** [*1] **Volkan Cevher** [1]

Abstract

Contrary to common belief, we show that gradient ascent-based unconstrained optimization methods frequently fail to perform machine unlearning, a phenomenon we attribute to the inherent statistical dependence between the forget and retain data sets. This dependence, which can manifest itself even as simple correlations, undermines the misconception that these sets can be independently manipulated during unlearning. We provide empirical and theoretical evidence showing these methods often fail precisely due to this overlooked relationship. For random forget sets, this dependence means that degrading forget set metrics (which, for the oracle, should mirror test set metrics) inevitably harms overall test performance. Going beyond random sets, we consider logistic regression as an instructive example where a critical failure mode emerges: inter-set dependence causes gradient descent-ascent iterations to progressively diverge from the oracle. Strikingly, these methods can converge to solutions that are not only far from the oracle but are potentially even further from it than the original model itself, rendering the unlearning process actively detrimental. A toy example further illustrates how this dependence can trap models in inferior local minima, inescapable via finetuning. Our theoretical insights are corroborated by experiments on complex neural networks, demonstrating that these methods do not perform as expected in practice due to this unaddressed statistical interplay.

1. Introduction

The widespread integration of large-scale machine learning models, especially in sensitive domains (e.g., medicine, cybersecurity), has spurred concerns over data privacy and model maintenance. Consequently, *machine unlearning*—selectively removing the influence of a chosen training example efficiently—has become a crucial ability (Ginart et al., 2019). Its applications are diverse, including managing outdated/toxic data, resolving copyright issues in generative models, and enhancing LLM alignment (Cao and Yang, 2015a; Pawelczyk et al., 2024; Liu, 2024; Li et al., 2024a).

The fundamental challenge in machine unlearning lies in designing efficient *unlearning algorithms* that do not degrade model performance.

Formally, given a model h_θ trained on dataset $\mathcal{D}$, unlearning aims to produce h_θ^{UL} after forgetting $\mathcal{F} \subset \mathcal{D}$, such that h_θ^{UL} approximates a model trained solely on the retain set $\mathcal{R} = \mathcal{D} \setminus \mathcal{F}$ (Cao and Yang, 2015b). Retraining from scratch on $\mathcal{R}$ is the ideal but often computationally prohibitive, especially for large datasets or frequent requests.

Many practical unlearning methods (Ginart et al., 2019; Kurmanji et al., 2024; Golatkar et al., 2020) employ fine-tuning heuristics, attempting to reverse the forget set's ($\mathcal{F}$) influence on the original model h_θ. These approaches, which we term Descent-Ascent (DA) algorithms, typically apply gradient ascent (Gradient Ascent) on $\mathcal{F}$ and gradient descent (Gradient Descent) on the retain set $\mathcal{R}$ for a few epochs (Kurmanji et al., 2023).

However, recent evaluations show DA approaches are often unreliable (Hayes et al., 2024; Kurmanji et al., 2024; Pawelczyk et al., 2023). They lack theoretical guarantees, clear stopping criteria, and are highly sensitive to fine-tuning hyperparameters like learning rate and duration.

This work identifies a crucial, often overlooked obstacle for DA methods: statistical dependencies between forget and retain set samples. We demonstrate that these dependencies can severely degrade unlearning performance, potentially causing complete failure, even in convex settings.

While we mainly study classification under the logistic loss, our findings on statistical dependencies should apply more broadly. Generative models such as LLMs also employ cross-entropy for next-token prediction, thus effectively performing a sequence of classification tasks. The inter-data dependencies we highlight would therefore pose analogous unlearning challenges for these models.

[*]Equal contribution [1]LIONS, EPFL, Lausanne, Switzerland. Correspondence to: ioannis.mavrothalassitis <@epfl.ch>.

Proceedings of the 42nd International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025. Copyright 2025 by the author(s).

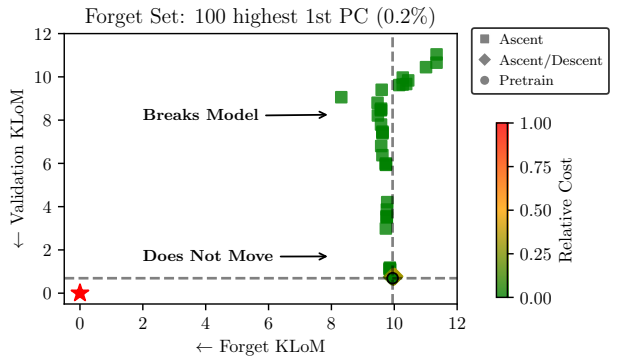

Figure 1: **Ascent Fails to Forget.** We apply Gradient Ascent and Gradient Descent/Ascent to Pretrained models to unlearn a selected forget set containing points of the first Principal Component (PC) of the influence matrix from Cifar-10. KLoM scores (x-axis, y-axis) measure the quality of unlearning on a given set by comparing the distribution distance between unlearned predictions and Oracle predictions (0 means perfect unlearning ★). We measure KLoM values over each data-point in a set and report the 95th percentile in each group. Different (x/y) points in the plot represent results for different unlearning method hyper-parameters. The colors indicate what is the relative cost of an unlearning method when compared to fully retraining the model. A Pretrained model (○) is similar to an Oracle on the validation set but very different on the forget set. On such set, unlearning with Gradient Ascent or Gradient Descent/Ascent either breaks the model or does not shift from the Pretrained starting point, consistently for most sets. Forget set selection and KLoM score metric follow Georgiev et al. (2024). Further details on method and evaluation hyper-parameters can be found in the Appendix.

Our main contributions can be summarized as follows: (i) We start by empirically showcasing that DA-based methods fail in practical settings under a robust evaluation and discuss limitations of previous methodologies. (ii) Supported by our empirical findings, we first show theoretically that unlearning random forget sets is impossible without causing model degradation, as unlearning random sets is equivalent in distribution to unlearning samples from the population data distribution. (iii) We move beyond forget and retain sets which share clear statistical dependencies to analyze the simple setting of multi-dimensional logistic regression, where we show inter-set correlations lead to DA failure modes. (iv) In our logistic regression analysis, we differentiate the impact of DA unlearning based on forget set size. We specifically show that for certain forget set sizes, DA can be harmful to the model, even when employing arbitrary early stopping. (v) Finally, using low-dimensional examples, we demonstrate how DA can lead the model to suboptimal local minima, which do not align with the minima achieved through retraining.

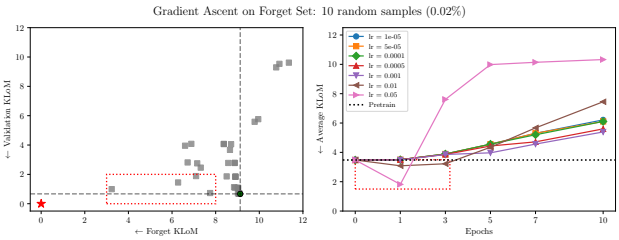

Figure 2: **The Ascent Forgets Illusion.** The left plot shows KLoM scores of Gradient Ascent when unlearning just 10 random samples (axis and points follow Fig. 1). Some runs (- - -) seem to achieve unlearning without breaking the model. On the right, we present the average KLoM between retain, validation and forget sets (y-axis) along time of unlearning (x-axis). We observe that in order for Gradient Ascent to unlearn such (easy) sets in practice, one would need to (i): select the learning rate, (ii) know when to stop fine-tuning.

Notation: We will use the following notation. We use uppercase bold letters for matrices $\boldsymbol{X} \in \mathbb{R}^{m \times n}$, lowercase bold letters for vectors $\boldsymbol{x} \in \mathbb{R}^m$ and lowercase letters for numbers $x \in \mathbb{R}$. Accordingly, the ith row and the element in the i, j position of a matrix $\boldsymbol{X}$ are given by $\boldsymbol{x}_i$ and x_{ij} respectively. We use the shorthand $[n] = \{1, \cdots, n\}$ for any natural number n. Let $\mathbb{1}_{(\cdot,\cdot)} : \mathbb{R} \times \mathbb{R} \to \{0, 1\}$ such that $\mathbb{1}_{(x,x)} = 1$, otherwise for $x \neq y, \mathbb{1}_{(x,y)} = 0$. We will denote our model with parameters θ as $h_\theta : \mathbb{R}^d \to \mathbb{R}$. We define a training dataset of size $|\mathcal{D}|$ as a set of samples and labels $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{|\mathcal{D}|} = \mathcal{D}$, composed of a "retain" set $\mathcal{R}$ and "forget" set $\mathcal{F}$ such that $|\mathcal{D}| = |\mathcal{R}| + |\mathcal{F}|$. We take "ascent" optimization on a sample to mean computing the gradient update w.r.t. to a loss $\nabla_\theta \ell$ and flipping its sign when updating the model parameters.

2. Ascent Methods Fail in the Wild

To rigorously evaluate unlearning efficacy, we use KLoM (KL Divergence of Margins (Georgiev et al., 2024)), measuring the distributional difference in classifier margins between 100 unlearned and 100 Oracle models. A KLoM score near zero indicates optimal unlearning.

Our main experiments evaluate two common gradient-based baselines: Gradient Ascent (GA), performing gradient ascent on the forget set, and Gradient Descent/Ascent (GDA), which adds retain set descent steps. We use ResNet-9 models on Cifar-10 (Krizhevsky, 2009) with varying forget sets. Fig. 1 illustrates that both GA and GDA often fail to significantly shift from the pretrained model or can severely degrade performance. These settings and results align with Georgiev et al. (2024).

These outcomes highlight a critical limitation in evaluating GA-based unlearning: the necessity for pre-defined hy-

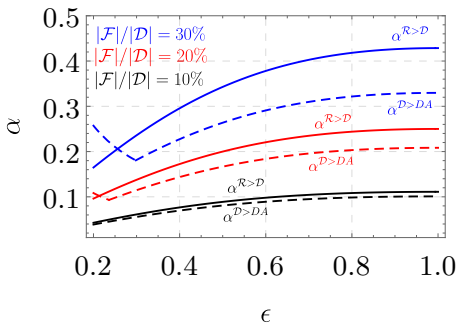

Figure 3: **Cross dimensional data correlations ϵ lead DA to failure for a certain range of values.** We present the range of α as a function of the correlation ϵ, for which we can guarantee that DA is detrimental. The (**- -**) lines represent the minimum α for which the coordinates of the original model become bigger than the coordinates of the DA unlearning algorithm and with the (**–**) the maximum α for which the coordinates of the oracle are bigger than those of the original model.

perparameter selection criteria, independent of final target metric performance, to avoid bias from instance-specific tuning. Without this, extensive hyperparameter search on small forget sets can create a misleading sense of successful unlearning, even with vanilla GA (Fig. 2). This issue is rooted in the "missing targets problem" (Hayes et al., 2024; Georgiev et al., 2024), i.e., the difficulty of defining a stopping criterion for GA-based optimization. Moreover, as different points unlearn at different rates (Georgiev et al., 2024), such a stopping value would likely need to be point-specific.

Motivated by these results, the following sections aim to demonstrate that the underlying statistical data dependencies may be a central cause for the typical failure modes of DA based unlearning methods.

3. Unlearning and Random Sets

A natural starting point for understanding how data correlations influence the unlearning process is that of random forget sets. In random forget sets, unlearning through DA is impossible. We state this formally in Lemma 1 for the Accuracy, but the same result holds for other metrics. The proof of Lemma 1 can be found in App. E.

Lemma 1 (Random Sets)**.** *Given a true distribution of samples $P_{\mathcal{T}}$ and a forget set $\mathcal{F}$ chosen uniformly at random from the dataset and a model with parameters θ, then the probability that the accuracy on the test set $Acc_{\mathcal{T}}$ and the forget set $Acc_{\mathcal{F}}$ diverge from one another by more than ϵ is upperbounded by the following inequality:*

$$P\left(|Acc_{\mathcal{T}} - Acc_{\mathcal{F}}| \geq \epsilon\right) \leq 2\exp\left(-2|\mathcal{F}|\epsilon^2\right).$$

It is important to note that a "good" unlearning method should not harm model quality under any given forget-set.

4. Models Diverge from Retraining Solutions Under DA Unlearning

Next, we extend our result under specific data correlations, without requiring statistical dependencies between the sets.

4.1. Logistic Regression

Here, we study binary logistic regression with a ridge parameter λ, and weights $\mathbf{w}$. Based on the work of Soudry et al. (2024), we use the exponential loss $\ell_i = e^{y_i h_\theta(\mathbf{x}_i)}$ as a more tractable proxy for the logistic loss. The pre-training ($\mathcal{D}$), retraining ($\mathcal{R}$) and GDA optimization methods (DA) minimize their respective losses

$$\begin{aligned}
\mathcal{L}_{\mathcal{D}} &= \frac{1}{|\mathcal{D}|}\sum_{i=1}^{\mathcal{D}} e^{-y_i\cdot\langle\mathbf{w},\mathbf{x}_i\rangle} + \frac{\lambda}{2}\|\mathbf{w}\|_2^2,\\
\mathcal{L}_{\mathcal{R}} &= \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle\mathbf{w},\mathbf{x}_i\rangle} + \frac{\lambda}{2}\|\mathbf{w}\|_2^2, \qquad (1)\\
\mathcal{L}_{\text{DA}} &= \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle\mathbf{w},\mathbf{x}_i\rangle} - \frac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} e^{-y_i\cdot\langle\mathbf{w},\mathbf{x}_i\rangle} + \frac{\lambda}{2}\|\mathbf{w}\|_2^2.
\end{aligned}$$

4.2. Semi-orthogonal Dataset Analysis

To analyze the detrimental effects of DA methods, we begin with a semi-orthogonal dataset. This setup, defined by the following two assumptions, allows us to study DA behavior under conditions of strong data correlation.

Assumption 1. *The data is separable into orthogonal sets S_j for each coordinate j.*

Assumption 2. *For any coordinate j and any sample i with $x_{i,j} \neq 0$, it holds that $y_i \cdot x_{i,j} = 1$.*

We discuss these assumptions and our proof techniques in App. C.1.1. This setting enables us to prove the following:

Lemma 2 (Divergence in Logistic Regression)**.** *Let $w_j^{(\mathcal{D})}$, $w_j^{(\mathcal{R})}$, and $w_j^{(DA)}$ be the jth coordinate of the convergence points for logistic regression on the original set $\mathcal{D}$, the retain set $\mathcal{R}$, and using the Descent-Ascent (DA) method, respectively. Then, for a specific range of α, we have: $\left(w_j^{(DA)} - w_j^{(\mathcal{D})}\right)\cdot\left(w_j^{(\mathcal{D})} - w_j^{(\mathcal{R})}\right) \geq 0$.*

Here, α represents the ratio of the forget set size to the retain set size within dimension j. The lemma holds for a specific range of α (detailed in App. H), notably including instances where $\alpha \leq |\mathcal{F}|/|\mathcal{R}|$, implying that for a one-dimensional dataset, the lemma's condition would **always** hold. We also highlight potential instabilities of DA methods:

Corollary 1. *As the ridge regularization parameter $\lambda \to 0$ and for $\alpha \to |\mathcal{F}|/|\mathcal{R}|$, we have that $\Delta_{\mathcal{R},\mathcal{D}} \to 0$ while $\Delta_{\mathcal{R},DA} \to \infty$.*

In the corollary, $\Delta_{\mathcal{R},\mathcal{D}}$ and $\Delta_{\mathcal{R},\text{DA}}$ denote the distance between the oracle (retrained on $\mathcal{R}$) solution and the original

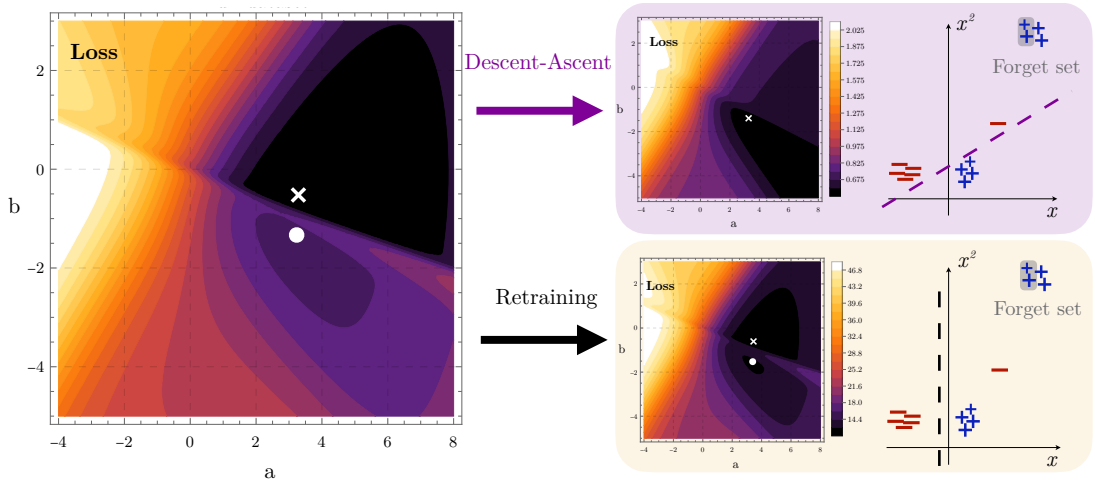

Figure 4: **Unlearning certain forget sets leads to the wrong decision boundary under GDA.** *Left:* We show the MSE loss landscape for a pretrained model on the problem described in App. B. We denote as ($\times$) the global minimum, while ($\circ$) is the local minimum. *Right:* The effective loss landscape observed in the GDA problem (top) and the retraining problem (bottom). The combination of these results shows that retraining keeps the model in the same global optimum as the pretrained model, while GDA chooses the local minimum. This is clearly manifest in the decision boundaries favored by the different methods, denoted in dashed lines. Next to the contour plots we present two dimensional illustrations of possible decision boundaries between the samples labeled as negative ($-$) and positive ($+$), while the forget set are the two positive points shaded in gray, as described in App. B. We show the decision boundaries for both GDA (right top) and retraining (right bottom).

($\mathcal{D}$) solution, and the oracle solution and the DA solution, respectively. This shows DA can diverge arbitrarily far from the desired retrained model.

4.3. Two-Dimensional Correlated Data

To investigate scenarios with more nuanced dependencies between forget and retain sets, we now consider a two-dimensional example, discussed in depth in App. C.1.2. We examine a case with two sample sets, S_i and S_j, characterized by feature vectors $x_k = (0, \dots, 0, 1, \epsilon, 0, \dots, 0)$ for samples $k \in S_i$ and $x_l = (0, \dots, 0, \epsilon, 1, 0, \dots, 0)$ for samples $l \in S_j$. Samples from S_i are designated to the retain set, while samples from S_j are in the forget set. In this setup, ϵ parameterizes the correlation between forget and retain samples, facilitating a parametric study of correlation's impact on DA performance. Analogous to the one-dimensional analysis, we define the forget-to-retain ratio within the relevant dimensions (i, j) as $|\mathcal{F}_{i,j}| = \alpha|\mathcal{R}_{i,j}|$.

Following the one-dimensional case, we show that for a reasonable α, it simultaneously holds that $(x^{\mathcal{R}} - x^{\mathcal{D}}) \cdot (x^{\mathcal{D}} - x^{\mathrm{DA}}) \geq 0$ and that $(y^{\mathcal{R}} - y^{\mathcal{D}}) \cdot (y^{\mathcal{D}} - y^{\mathrm{DA}}) \geq 0$.

Lemma 3. *For $\alpha \geq \alpha^{\mathcal{D}>DA} = \max\left\{\alpha_x^{\mathcal{D}>DA}, \alpha_y^{\mathcal{D}>DA}\right\}$ we have that $x^{\mathcal{D}} \geq x^{DA}$ and that $y^{\mathcal{D}} \geq y^{DA}$.*

Lemma 4. *For $\alpha \leq \alpha^{\mathcal{R}>\mathcal{D}} = \min\left\{\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}\right\}$ we have that $x^{\mathcal{R}} \geq x^{\mathcal{D}}$ and that $y^{\mathcal{R}} \geq y^{\mathcal{D}}$, with $\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}$.*

We find a range for α where DA proves detrimental, as illustrated in Fig. 3. See App. C.1.2 for the full derivation.

5. Convergence to "Bad" Minima

Beyond demonstrating DA methods' immediate detrimental impact, we address a further critical concern: *Can subsequent finetuning on the retain set remedy these harmful effects?* Unfortunately, for non-convex models such as neural networks, the answer is often no. This is illustrated in Fig. 4, with a detailed instructive example given in App. B. The core issue, once again, lies in the statistical dependencies between the forget and retain sets. If forgotten samples are significantly correlated with retained samples crucial for defining optimal decision boundaries, the DA process can steer the model into suboptimal local minima. These minima may be inescapable through standard finetuning on the retain set, leading to persistently degraded model performance.

6. Conclusions

While our findings highlight significant challenges for current ascent-based unlearning, we believe that they are instructive for the development of more robust methods. The weaknesses we identify primarily stem from ascent steps neglecting the data dependencies between the forget and retain sets. Future research into ascent-based unlearning should therefore explicitly account for these inter-set relationships. Furthermore, our work suggests that alternative approaches, such as rewinding techniques (Mu and Klabjan, 2024) or stochastic methods (Chien et al., 2024), may offer more reliable unlearning, particularly when dataset properties and their internal correlations are unknown.

References

Antonio Ginart, Melody Y. Guan, Gregory Valiant, and James Zou. Making ai forget you: Data deletion in machine learning. In *Proceedings of the 33rd Conference on Neural Information Processing Systems (NeurIPS 2019)*, 2019.

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

Martin Pawelczyk, Jimmy Z Di, Yiwei Lu, Gautam Kamath, Ayush Sekhari, and Seth Neel. Machine unlearning fails to remove data poisoning attacks. *arXiv preprint arXiv:2406.17216*, 2024.

Ken Ziyu Liu. Machine unlearning in 2024, Apr 2024. URL https://ai.stanford.edu/~kzliu/blog/unlearning.

Nathaniel Li, Alexander Pan, Anjali Gopal, Summer Yue, Daniel Berrios, Alice Gatti, Justin D. Li, Ann-Kathrin Dombrowski, Shashwat Goel, Long Phan, Gabriel Mukobi, Nathan Helm-Burger, Rassin Lababidi, Lennart Justen, Andrew B. Liu, Michael Chen, Isabelle Barrass, Oliver Zhang, Xiaoyuan Zhu, Rishub Tamirisa, Bhrugu Bharathi, Adam Khoja, Zhenqi Zhao, Ariel Herbert-Voss, Cort B. Breuer, Samuel Marks, Oam Patel, Andy Zou, Mantas Mazeika, Zifan Wang, Palash Oswal, Weiran Lin, Adam A. Hunt, Justin Tienken-Harder, Kevin Y. Shih, Kemper Talley, John Guan, Russell Kaplan, Ian Steneker, David Campbell, Brad Jokubaitis, Alex Levinson, Jean Wang, William Qian, Kallol Krishna Karmakar, Steven Basart, Stephen Fitz, Mindy Levine, Ponnurangam Kumaraguru, Uday Tupakula, Vijay Varadharajan, Ruoyu Wang, Yan Shoshitaishvili, Jimmy Ba, Kevin M. Esvelt, Alexandr Wang, and Dan Hendrycks. The wmdp benchmark: Measuring and reducing malicious use with unlearning. In *International Conference on Machine Learning (ICML)*, 2024a.

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

Kristian Georgiev, Roy Rinberg, Sung Min Park, Shivam Garg, Andrew Ilyas, Aleksander Madry, and Seth Neel. Attribute-to-delete: Machine unlearning via datamodel matching, 2024. URL https://arxiv.org/abs/2410.23232.

Meghdad Kurmanji, Peter Triantafillou, and Eleni Triantafillou. Towards unbounded machine unlearning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Aditya Golatkar, Alessandro Achille, and Stefano Soatto. Eternal sunshine of the spotless net: Selective forgetting in deep networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9304–9312, 2020.

Meghdad Kurmanji, Peter Triantafillou, Jamie Hayes, and Eleni Triantafillou. Towards unbounded machine unlearning, 2023. URL https://arxiv.org/abs/2302.09880.

Jamie Hayes, Ilia Shumailov, Eleni Triantafillou, Amr Khalifa, and Nicolas Papernot. Inexact unlearning needs more careful evaluations to avoid a false sense of privacy, 2024.

Martin Pawelczyk, Seth Neel, and Himabindu Lakkaraju. In-context unlearning: Language models as few shot unlearners, 2023.

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

Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data, 2024. URL https://arxiv.org/abs/1710.10345.

Siqiao Mu and Diego Klabjan. Rewind-to-delete: Certified machine unlearning for nonconvex functions. *arXiv preprint arXiv:2409.09778*, 2024.

Eli Chien, Haoyu Wang, Ziang Chen, and Pan Li. Langevin unlearning: A new perspective of noisy gradient descent for machine unlearning. *arXiv preprint arXiv:2401.10371*, 2024.

Yinjun Wu, Edgar Dobriban, and Susan Davidson. Deltagrad: Rapid retraining of machine learning models. In *International Conference on Machine Learning (ICML)*, 2020.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *Proceedings of the 32nd International Conference on Algorithmic Learning Theory (ALT)*, 2021.

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

Shauli Ravfogel, Michael Twiton, Yoav Goldberg, and Ryan D Cotterell. Linear adversarial concept erasure. In *International Conference on Machine Learning (ICML)*, 2022.

Ronen Eldan and Mark Russinovich. Who's harry potter? approximate unlearning in llms. *arXiv preprint arXiv:2310.02238*, 2023.

Nupur Kumari, Bingliang Zhang, Sheng-Yu Wang, Eli Shechtman, Richard Zhang, and Jun-Yan Zhu. Ablating concepts in text-to-image diffusion models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Zexuan Zhong, Zhengxuan Wu, Christopher D Manning, Christopher Potts, and Danqi Chen. Mquake: Assessing knowledge editing in language models via multi-hop questions. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2023.

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

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

Fazl Barez, Tingchen Fu, Ameya Prabhu, Stephen Casper, Amartya Sanyal, Adel Bibi, Aidan O'Gara, Robert Kirk, Ben Bucknall, Tim Fist, Luke Ong, Philip Torr, Kwok-Yan Lam, Robert Trager, David Krueger, Sören Mindermann, José Hernandez-Orallo, Mor Geva, and Yarin Gal. Open problems in machine unlearning for ai safety, 2025. URL https://arxiv.org/abs/2501.04952.

Jinghan Jia, Jiancheng Liu, Parikshit Ram, Yuguang Yao, Gaowen Liu, Yang Liu, Pranay Sharma, and Sijia Liu. Model sparsity can simplify machine unlearning, 2024. URL https://arxiv.org/abs/2304.04934.

Guihong Li, Hsiang Hsu, Chun-Fu Chen, and Radu Marculescu. Fast-ntk: Parameter-efficient unlearning for large-scale models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 227–234, 2024b.

Alexander Warnecke, Lukas Pirch, Christian Wressnegger, and Konrad Rieck. Machine unlearning of features and labels. *arXiv preprint arXiv:2108.11577*, 2021.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015. URL http://arxiv.org/abs/1512.03385.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Shibani Santurkar, Dimitris Tsipras, and Aleksander Madry. Breeds: Benchmarks for subpopulation shift. In *International Conference on Learning Representations (ICLR)*, 2021.

A. Related Work

Machine unlearning: Unlearning methods can be used to either remove particular samples (Cao and Yang, 2015b; Ginart et al., 2019; Wu et al., 2020; Neel et al., 2021; Bourtoule et al., 2021), or to remove subsets of the data which share certain underlying features, captured by abstract concepts (Ravfogel et al., 2022; Eldan and Russinovich, 2023; Kumari et al., 2023; Zhong et al., 2023). In this work, we focus on the prior, though we believe some of our results may be extended to the latter setting. Exact unlearning methods (Bourtoule et al., 2021) offer theoretical guarantees but often sacrifice accuracy, leading to widespread adoption of approximate methods in deep learning. These approximate approaches are evaluated through membership inference attacks (Golatkar et al., 2020; Goel et al., 2022; Hayes et al., 2024) and backdoor removal capabilities (Pawelczyk et al., 2024). As Thudi et al. (2022) note, meaningful evaluation must focus on algorithmic behavior rather than individual models due to deep learning's stochastic nature. For a review of open problems in machine unlearning, see (Barez et al., 2025) and references therein.

Unlearning approaches in deep learning. Current approaches primarily use gradient-based methods, including partial fine-tuning (Goel et al., 2022), AD combinations (Kurmanji et al., 2024), and sparsity-regularized fine-tuning (Jia et al., 2024). Alternative methods employ local quadratic approximations (Golatkar et al., 2020; Li et al., 2024b) or influence functions (Warnecke et al., 2021). One of the most used unlearning methods, SCRUB (Hayes et al., 2024) fine-tunes models using KL divergence objectives, but faces similar underlying challenges as other methods. The approach presented in Georgiev et al. (2024) introduces a predictive data attribution approach with good unlearning quality under a robust evaluation, although it raises some scalability concerns if we account for the full cost the method. In this work we focus on DA based methods.

B. Low dimensions: Descent-Ascent Favors The wrong solutions

While our previous theoretical analysis demonstrates that DA methods can be harmful to the model, it fails to demonstrate a final concern about these methods, we would like to raise. *Is it possible to remedy the harmful effects of these methods through finetuning on the retain afterwards?*

The answer that we give to this question unfortunately is not always, for neural networks or in general non-convex function classes. To demonstrate this let us consider a binary classification problem using a two dimensional kernel, with labels $y_i \in \{-1, 1\}$, data composed of $\mathbf{x}_i = (x_i, x_i^2)$ and Mean Squared Error (MSE) loss with ridge regularization $\lambda \in \mathbb{R}^+$. The network is taken to be a sigmoidal network with two parameters $\theta = (a, b)$, such that its output is $h_\theta(\mathbf{x}_i) = \sigma(ax_i + bx_i^2)$, where $\sigma(z) = 1/(1 + e^{-(1+z)/2})$.

We choose 4 samples in the configuration: $\mathcal{D} = \{\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4\} = \{(-1, 1), (1, 1), (3, 9), (4, 16)\}$, with labels $\{y_1, y_2, y_3, y_4\} = \{-1, 1, -1, 1\}$, respectively. In order to model the effect of multiple points clustered together, we give each point a different weight in the loss function, such that

$$\mathcal{L} = \frac{1}{|\mathcal{D}|} \sum_{i=1}^{|\mathcal{D}|} \alpha_i \ell_i + \frac{\lambda}{2} \|\theta\|_2^2, \tag{1}$$

where ℓ_i are the single sample loss functions, and $\{\alpha_1, \alpha_2, \alpha_3, \alpha_4\} = \{5, 4, 1, 4\}$ represent the number of points clustered together, as illustrated in Fig. 4, where $\lambda = 0.1$. This means that the effective number of points that the classifier sees is $\sum_i \alpha_i$. The data configuration is chosen to illustrate the failure mode of DA, while the dataset selection is arbitrary the key mode of failure is the high correlation between the forget set and a subset of the retain.

Suppose we would like to unlearn two of the positive samples positioned at $\mathbf{x}_4$. Retraining would correspond to simply setting $\alpha_4 = 2$, and applying gradient descent. Notice that this provides little to no change for the minima location and the contour lines between the original dataset $\mathcal{D}$ and the retraining set $\mathcal{R}$. In contrast, Performing GDA would amount to setting $\alpha_4 = 0$, since two points will contribute the exact opposite gradient as the other two at the same position, effectively erasing them.

We find that this example can be simply understood by counting arguments: since the original dataset contains effectively 6 negative samples and 8 positive samples, the optimal decision boundary is given by the separating plane which correctly classifies the largest number of samples.

The pretrained model is optimal when $\mathbf{x}_1, \mathbf{x}_2$ and $\mathbf{x}_4$ are correctly classified, while mislabeling $\mathbf{x}_3$ (13 correct, 1 incorrect). Retraining simply reduces the weight of $\mathbf{x}_4$, and keeping the same plane is still preferential (11 correct, 1 incorrect).

However, performing GDA sets the gradients of half of the points at $\mathbf{x}_4$ to cancel the other half, so it optimal to re-orient the decision boundary so that all samples are correctly classified (10 correct, 0 incorrect), while in reality, the algorithm has been tricked into finding a suboptimal solution (10 correct, 2 incorrect).

The qualitative analysis of this two-dimensional example shows that certain choices of forget sets that are highly correlated to the retain can lead to irreversible model degradation when using DA.

C. Derivation DA for high dimensional logistic regression

C.1. High Dimensions: Correlated Data Causes Diverging Solutions in Logistic Regression

C.1.1. Data Correlations on a Single Dimension

We start from the case of a semi orthogonal dataset. Using the following assumptions: These assumptions correspond to a dataset in d dimensions where there are sets of samples on orthogonal axes to one another. As a result, data points that lie in different sets S_j, are perfectly orthogonal and uncorrelated; however, data points that lie in the same set are fully correlated with one another.

Recall our hypothesis that data dependencies can cause DA methods to degrade model metrics, instead of converging to an oracle model, we will pick a subset of a set S_j as our forget set. This will allow for a simplistic analysis while testing the hypothesis for a highly correlated forget set.

Let $|\mathcal{R}_j|$ the size of the retain set for samples with $x_{i,j} \neq 0$, then in order to model the behavior of the minimizers of Eq. (1), for forget sets of different sizes, we define the jth forget set fraction size as $|\mathcal{F}_j| = \alpha \cdot |\mathcal{R}_j|$. A simple example of this setting can be a set of retain points of $x_j = 1, y_j = 1$ and a set of forget points of $x_j = -1, y_j = -1$, where we are practically requested to remove all (or some) of the negative samples. The effect of unlearning a forget set on a particular coordinate axis j, can then be shown to obtain closed form solutions as given by Lemma 5, proven in App. F.2.

Lemma 5 (Closed Form)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ and w_j^{DA} be the jth coordinate of **any** local minima/maxima for the logistic regression problems defined in Eq. (1), then they admit the form:*

$$w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right), w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right), w_j^{DA} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right),$$

where $W(z)$ corresponds to the Lambert-W function, the solution to $z = W(z)e^{W(z)}$.

It follows directly from Lemma 5, that by changing the value of α, which determines the ratio of the size of the forget set to that of the retain in this coordinate, the solutions will be ordered by their magnitude. Concretely, Lemma 2 shows that the DA solution is always **farthest** away from the oracle solution, while the oracle and pre-trained solutions remain close and more importantly the DA solution and the oracle solution lie in opposite directions with respect to the initial solution of pre-training $w_j^{\mathcal{D}}$. This observation implies that performing DA in this setup always converges away from the oracle solution, thus doing nothing at all is a better strategy than DA. The aforementioned observation can be formally decomposed in the following lemmas. We prove Lemma 2 in App. H, which gives a formal statement regarding the fact that the minima of DA and the oracle are in opposite directions with respect to the minimum of the intial dataset $\mathcal{D}$.

Lemma 2 (Divergence in Logistic Regression)**.** *Let $w_j^{(\mathcal{D})}$, $w_j^{(\mathcal{R})}$, and $w_j^{(DA)}$ be the jth coordinate of the convergence points for logistic regression on the original set $\mathcal{D}$, the retain set $\mathcal{R}$, and using the Descent-Ascent (DA) method, respectively. Then, for a specific range of α, we have: $\left(w_j^{(DA)} - w_j^{(\mathcal{D})}\right) \cdot \left(w_j^{(\mathcal{D})} - w_j^{(\mathcal{R})}\right) \geq 0$.*

We defer the reader to App. H for the exact range of α, for which Lemma 2 holds, let us point out that the lemma holds for $\alpha \leq |\mathcal{F}|/|\mathcal{R}|$, this means that if we were working on a purely 1 dimensional dataset, this lemma would **always** hold. Lemma 2 answers our original question of whether data correlations cause DA methods to harm the model in the positive. Before proceeding to the study of higher dimensions, we would like to comment on the stability of the process of unlearning under DA methods.

Stability of DA methods: We begin by characterizing the distance between the different stationary points for the three problems.

Lemma 6 provides an upperbound on the distance between the oracle solution and the initial solution for $\mathcal{D}$. Its counterpart,

Lemma 7 provides a lower bound on the distance between the oracle solution and the DA solution. The proof for Lemma 6 can be found in App. I, while the proof for Lemma 7 lies in App. J

Lemma 6 (Distance Growth)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ the jth coordinate of the convergence point for the logistic regression problem for the original set $\mathcal{D}$ and the retain set $\mathcal{R}$ respectively. It holds that the distance $\Delta_{\mathcal{R},\mathcal{D}} = |w_j^{\mathcal{D}} - w_j^{\mathcal{R}}| \leq \left|\ln\left((1+\alpha)\frac{|\mathcal{R}|}{|\mathcal{D}|}\right)\right|$, for any value of $\lambda > 0$.*

Lemma 7 (Distance Unlearning)**.** *Let $w_j^{\mathcal{R}}, w_j^{DA}$ the jth coordinate of the convergence point for the logistic regression problem for the retain set $\mathcal{R}$ and the Descent Ascent method respectively. It holds that for for $\alpha \geq |\mathcal{F}|/|\mathcal{R}|$ the distance $\Delta_{\mathcal{R},DA} = |w_j^{\mathcal{R}} - w_j^{DA}| \geq W_0\left(|\mathcal{R}_j|/(\lambda|\mathcal{R}|)\right)$*

Employing Lemma 6 and Lemma 7, one can derive the following Corollary.

Corollary 2. *As the ridge $\lambda \to 0$ for $\alpha \to |\mathcal{F}|/|\mathcal{R}|$, we have that $\Delta_{\mathcal{R},\mathcal{D}} \to 0$ and $\Delta_{\mathcal{R},DA} \to \infty$.*

Cor. 1 demonstrates how unlearning using DA is very volatile and even a few steps of the method can cause the model to diverge.

A possible stabilization effect of iterative DA: So far, we have focused on the behavior of minimizers of Eq. (1), which describes a simultaneous descent-ascent algorithm. In practice, however, iterative methods are typically used, where one first performs a step of ascent on the forget set, followed descent on the retain set. In App. F.3, we show that for small learning rates $\eta \to 0$, the iterative method is nearly identical to the simultaneous update. Namely the derivative used for the update rule is

$$w_j^{t+1} \leftarrow w_j^t - \eta\left(-\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} + \frac{\alpha \cdot |\mathcal{R}_j|}{|\mathcal{F}|}e^{-w_j^t} + 2\lambda w_j^t\right),$$

where the only difference is a factor of 2 in front of the regularization that differs from the normal DA loss. We have omitted a term which is of the order of $\mathcal{O}\left(\eta^2\right)$, since the solution, should it exist has w_j^t small and a term with $\eta^2 \to 0$ has negligable contribution.

The leading correction term $\mathcal{O}\left(\eta^2\right)$ which was omitted in the update rule above stops the algorithms solution w_j^{DA} from diverging, since the term is of the form

$$\eta^2\alpha\frac{|\mathcal{R}_j|^2}{|\mathcal{F}||\mathcal{R}|}e^{-2w_j^t} - \eta^2\lambda w_j^t\alpha\frac{|\mathcal{R}_j|}{|\mathcal{F}|}e^{-w_j^t},$$

which increases for larger w_j^t. This addresses stability concerns; however, it does nothing to remedy our main concern raised in Lemma 2 regarding the harmful effect of these methods on the model.

C.1.2. Cross Dimensional Data Correlations

In the previous section we studied the case where our samples are fully correlated, since they existed in a single dimension. In this section we will consider the two dimensional case where we have two sets of samples S_i and S_j, which have values $x_i = (0, \ldots, 0, 1, \epsilon, 0, \ldots, 0)$ and $x_j = (0, \ldots, 0, \epsilon, 1, 0, \ldots, 0)$ respectively. We will consider the case where the samples of S_i are all in the retain set, while the samples of S_j are all in the forget. In this case the correlation between the samples in the forget and the retain set depends on ϵ and therefore this allows us to do a parametric study of the effect of correlation between the forget and the retain on the performance of DA based methods. In similar fashion to the 1 dimensional case we will consider that the forget set $|\mathcal{F}_{i,j}| = \alpha|\mathcal{R}_{i,j}|$, where $F_{i,j}, R_{i,j}$ the forget and the retain over the i, j dimensions, respectively. In order to facilitate the analysis we will change the coordinate system only for the i and the j coordinate to $x = w_i + \epsilon w_j$ and $y = w_i\epsilon + w_j$. Let $x^{\mathcal{R}}, y^{\mathcal{R}}$ the coordinates for the oracle model stationary point, $x^{\mathcal{D}}, y^{\mathcal{D}}$ for the pretrain model and $x^{\mathrm{DA}}, y^{\mathrm{DA}}$ for the DA unlearning scheme, we can give the following characterizations:

Lemma 8. *The closed form solution for the stationary points for the retrain set is given as:*

$$x^{\mathcal{R}} = \mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{R}|}\right), \quad y^{\mathcal{R}} = \frac{2\epsilon}{1+\epsilon^2}\mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{R}|}\right).$$

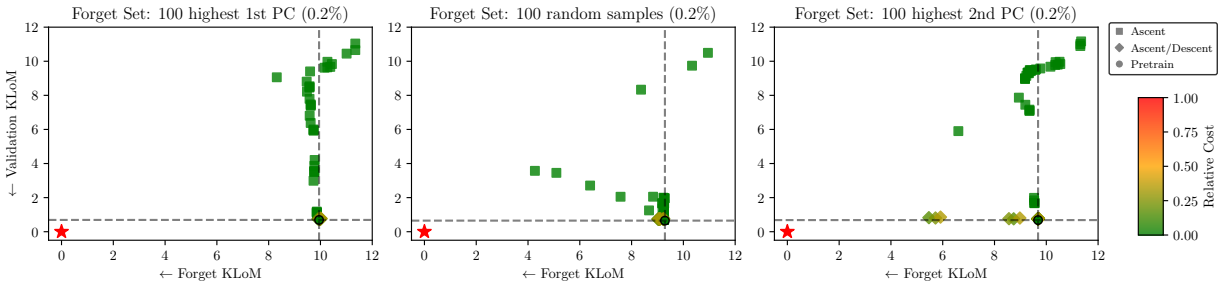

Figure 5: **Different Unlearning Difficulties.** We present the KLoM scores of Gradient Ascent and Gradient Descent/Ascent when unlearning over different forget sets (axes and points follow Fig. 1). In general, the majority of runs either do nothing or break the model. Empirically, we find highly important points (left) to be the hardest to unlearn with zero realizations showing any unlearning signs at all. Random samples (center) show some Gradient Ascent runs improving the forget KLoM but with significant degradation in the models. Finally, for a set with second PC points (right) we observe some Gradient Descent/Ascent runs improve the forget KLoM without breaking the model but at a high cost, around 25% of retraining an Oracle for unlearning 0.2% of the data.

Lemma 9. *For the stationary points of the original set, one can derive the following ranges.*

$$\begin{aligned} \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)+2\alpha\epsilon)\right) &\leq x^{\mathcal{D}} \leq \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right)+\mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right), \\ \mathrm{W}\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right) &\leq y^{\mathcal{D}} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon+\alpha(1+\epsilon^2))\right). \end{aligned}$$

Lemma 10. *For the stationary points of the model trained by DA methods we can derive the following ranges.*

$$x^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)-\frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon\right), \qquad y^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon-\frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)\right).$$

While the problem becomes more complex in this case and to our knowledge it is not possible to compute an exact solution, the above Lemmas provide enough information for our purpose. The proofs for all of these Lemmas can be found in App. K.1. In similar fashion to the 1 dimensional case we would like to show that there exists a reasonable α, for which we have that $(x^{\mathcal{R}}-x^{\mathcal{D}})\cdot(x^{\mathcal{D}}-x^{\mathrm{DA}})\geq 0$ and at the same time $(y^{\mathcal{R}}-y^{\mathcal{D}})\cdot(y^{\mathcal{D}}-y^{\mathrm{DA}})\geq 0$.

Lemma 11. *For* $\alpha \geq \alpha^{\mathcal{D}>DA} = \max\left\{\frac{1+\epsilon^2}{2\epsilon}\frac{|\mathcal{F}|^2}{|\mathcal{R}|(|\mathcal{D}|+|\mathcal{F}|)}, \frac{2\epsilon}{1+\epsilon^2}\frac{|\mathcal{F}||\mathcal{D}|}{|\mathcal{R}|(|\mathcal{D}|+|\mathcal{F}|)}\right\}$ *we have that* $x^{\mathcal{D}} \geq x^{DA}$ *and that* $y^{\mathcal{D}} \geq y^{DA}$.

Lemma 4. *For* $\alpha \leq \alpha^{\mathcal{R}>\mathcal{D}} = \min\left\{\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}\right\}$ *we have that* $x^{\mathcal{R}} \geq x^{\mathcal{D}}$ *and that* $y^{\mathcal{R}} \geq y^{\mathcal{D}}$*, with* $\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}$.

We omit the exact values of $\alpha_x^{\mathcal{R}>\mathcal{D}}$ and $\alpha_y^{\mathcal{R}>\mathcal{D}}$, which can be found in App. K.2 along with the proofs for Lemma 11 and Lemma 4. Since the range of ϵ for which $(x^{\mathcal{R}}, y^{\mathcal{R}}) \geq (x^{\mathcal{D}}, y^{\mathcal{D}}) \geq (x^{\mathrm{DA}}, y^{\mathrm{DA}})$ cannot be resolved analytically, we show numerically in Fig. 3 that this range is typically large, and broadens as the fraction of samples to be forgotten increases, while the relevant window of correlation strength ϵ is wider for smaller correlation.

D. Additional discussion on Experimental results

We also observe that the difficulty of unlearning varies greatly depending on the specific forget set selected, as shown in Fig. 5. In general, we find GA and GDA methods to be fragile. The extreme sensitivity to hyperparameters, unclear stopping criteria for Gradient Ascent, and substantial computational costs in using Gradient Descent on the retain set to fix models, severely restrict their practicality. Fundamentally, performing gradient ascent on individual points is not aligned with the core definition of unlearning, making these approaches unsuitable for reliable and consistent machine unlearning in real-world scenarios. In the Appendix, we include the methodology details for forget sets, KLoM, hyperparameters along with additional results on more forget sets, models (ResNet-18 (He et al., 2015)) and datasets (ImageNetLiving-17 (Deng et al., 2009; Santurkar et al., 2021)).

E. Proof of Lemma for Random Sets

In this section we provide proof that for a forget set, selected uniformly at random from the dataset it is with high probability impossible to differentiate the accuracy, loss, or any other metric between the test and the forget set, given that both of them are large enough. In this section we provide the proof for the accuracy metric, but for other metrics the proof follows in like manner. Intuitively this stems from the fact that for a model which has "unlearned" a forget set, that set is a random set for it.

We will use the following notation. Let $\mathbb{1}_{(\cdot,\cdot)} : \mathbb{R} \times \mathbb{R} \to \{0, 1\}$ such that $\mathbb{1}_{(x,x)} = 1$, otherwise for $x \neq y, \mathbb{1}_{(x,y)} = 0$. We will denote our model with parameters θ as $h_\theta : \mathbb{R}^d \to \mathbb{R}$.

Lemma 1 (Random Sets)**.** *Given a true distribution of samples $P_\mathcal{T}$ and a forget set $\mathcal{F}$ chosen uniformly at random from the dataset and a model with parameters θ, then the probability that the accuracy on the test set $Acc_\mathcal{T}$ and the forget set $Acc_\mathcal{F}$ diverge from one another by more than ϵ is upperbounded by the following inequality:*

$$P\left(|Acc_\mathcal{T} - Acc_\mathcal{F}| \geq \epsilon\right) \leq 2\exp\left(-2|\mathcal{F}|\epsilon^2\right).$$

Proof. For each sample (x_i, y_i), we calculate the correct response on that sample, as $\mathbb{1}_{(h_\theta(x_i),y_i)}$, consequently the response of the model for any sample is an independent rendom variable. So we get the following random variables, which correspond to the accuracy of the model on the forget set $\mathcal{F}$ and the test set $\mathcal{T}$ respectively.

$$\text{Acc}_\mathcal{T} = \mathbb{E}_{(x_i,y_i)\sim P_\mathcal{T}}\left[\mathbb{1}_{(h_\theta(x_i),y_i)}\right]$$

$$\text{Acc}_\mathcal{F} = \frac{1}{|\mathcal{F}|}\sum_{(x_i,y_i)\in\mathcal{F}} \mathbb{1}_{(h_\theta(x_i),y_i)}$$

In order to proceed we will utilize Hoeffding's Inequality, which we state below for completeness:

Lemma 12. *Let $Z_1, Z_2, \ldots, Z_n$ be independent random variables such that $Z_i \in [a_i, b_i]$. Define their sum as:*

$$S_n = \sum_{i=1}^{n} Z_i$$

and let $\mathbb{E}[S_n]$ be the expected value of S_n. Then, for any $t > 0$, the following bound holds:

$$P\left(|S_n - \mathbb{E}[S_n]| \geq nt\right) \leq 2\exp\left(\frac{-2n^2t^2}{\sum_{i=1}^{n}(b_i - a_i)^2}\right)$$

In our case we have that $\frac{1}{n}S_n = \text{Acc}_\mathcal{F}$. Since the Forget set $\mathcal{F}$ is selected uniformly at random, we have that:

$$\begin{aligned}
\mathbb{E}\left[\text{Acc}_\mathcal{F}\right] &= \mathbb{E}\left[\frac{1}{|\mathcal{F}|}\sum_{(x_i,y_i)\in\mathcal{F}} \mathbb{1}_{(h_\theta(x_i),y_i)}\right] \\
&= \frac{1}{|\mathcal{F}|}\sum_{(x_i,y_i)\in\mathcal{F}} \mathbb{E}_{(x_i,y_i)\sim P_\mathcal{T}}\left[\mathbb{1}_{(h_\theta(x_i),y_i)}\right] \\
&= \mathbb{E}_{(x_i,y_i)\sim P_\mathcal{T}}\left[\mathbb{1}_{(h_\theta(x_i),y_i)}\right] \\
&= \text{Acc}_\mathcal{T}
\end{aligned}$$

Since the random variables $\mathbb{1}_{(h_\theta(x_i),y_i)} \in [0, 1]$, we have that:

$$P\left(|\text{Acc}_\mathcal{T} - \text{Acc}_\mathcal{F}| \geq \epsilon\right) \leq 2exp\left(-2|\mathcal{F}|\epsilon^2\right)$$

which gives the lemma statement. □

The above lemma gives a formal statement, as to why maximizing the error on random forget sets does not correspond to true unlearning, since the metrics in the forget set should match those in the test set.

F. Logistic Regression

F.1. Problem Statement

The logistic regression problem for the full dataset $\mathcal{D}$, retain set $\mathcal{R}$ and for the Descent-Ascent algorithm can be restated as:

$$\begin{aligned}
\textbf{minimization } \mathcal{D}: & \frac{1}{|\mathcal{D}|}\sum_{i=1}^{\mathcal{D}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2 \\
\textbf{minimization } \mathcal{R}: & \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2 \\
\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: & \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} - \frac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2
\end{aligned} \tag{2}$$

F.2. Single Dimension

In this section, we compare the solutions of training a logistic regression model on a full dataset $\mathcal{D}$, purely on the retain set $\mathcal{R}$ and doing GDA on the forget set $\mathcal{F}$. We will also include a regularization term. The corresponding objective functions would be:

We can derivate the above to get the following equations for their solutions respectively.

$$\begin{aligned}
&(\text{minimization } \mathcal{D}) && \tfrac{1}{|\mathcal{D}|}\textstyle\sum_{i=1}^{\mathcal{D}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0 \\
&(\text{minimization } \mathcal{R}) && \tfrac{1}{|\mathcal{R}|}\textstyle\sum_{i=1}^{\mathcal{R}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0 \\
&(\text{Descent } \mathcal{R} - \text{Ascent } \mathcal{F}) && \tfrac{1}{|\mathcal{R}|}\textstyle\sum_{i=1}^{\mathcal{R}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} - \tfrac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0
\end{aligned}$$

So we can express each coordinate j of the minimizer for the three cases, as:

$$\begin{aligned}
&(\text{minimization } \mathcal{D}) && w_j = \tfrac{1}{\lambda|\mathcal{D}|}(\textstyle\sum_{i=1}^{\mathcal{D}} y_i\cdot x_{i,j} e^{-y_i\cdot\langle w,x_i\rangle}) \\
&(\text{minimization } \mathcal{R}) && w_j = \tfrac{1}{\lambda|\mathcal{R}|}(\textstyle\sum_{i=1}^{\mathcal{R}} y_i\cdot x_{i,j} e^{-y_i\cdot\langle w,x_i\rangle}) \\
&(\text{Descent } \mathcal{R} - \text{Ascent } \mathcal{F}) && w_j = \tfrac{1}{\lambda|\mathcal{R}|}(\textstyle\sum_{i=1}^{\mathcal{R}} y_i\cdot x_{i,j} e^{-y_i\cdot\langle w,x_i\rangle}) - \tfrac{1}{\lambda|\mathcal{F}|}(\textstyle\sum_{i=1}^{\mathcal{F}} y_i\cdot x_{i,j} e^{-y_i\cdot\langle w,x_i\rangle})
\end{aligned}$$

F.3. Iterating Gradient Descent and Ascent

Here, we consider the iterative gradient descent-ascent algorithm, where we first perform a gradient descent step on the retain set, followed by a gradient ascent step on the forget set. We show that to leading order in the small learning rate expansion, the solution found by iterative GA is identical to the one given by GA in Eq. (2). For iterative GA, the dynamics are given by

$$\begin{aligned}
w_j^{t+1} &= w_j^t + \eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - \lambda w_j^t\right), \\
w_j^{t+2} &= w_j^{t+1} - \eta\left(\frac{\epsilon\cdot|\mathcal{R}_j|}{|\mathcal{F}|}e^{-w_j^{t+1}} + \lambda w_j^{t+1}\right),
\end{aligned} \tag{3}$$

where η is the learning rate for both steps. Plugging in the result of w_j^{t+1} into the expression for w_j^{t+2} and expanding for small $\eta \ll 1$, we obtain the following update rule

$$\begin{aligned} w_j^{t+2} &= w_j^t + \eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - \lambda w_j^t\right) \\ &- \eta\left(\frac{\epsilon\cdot|\mathcal{R}_j|}{|\mathcal{F}|}e^{-\left(w_j^t+\eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t}-\lambda w_j^t\right)\right)} + \lambda\left(w_j^t + \eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - \lambda w_j^t\right)\right)\right) \\ &\simeq w_j^t + \eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - 2\lambda w_j^t\right) - \eta\left(\frac{\epsilon\cdot|\mathcal{R}_j|}{|\mathcal{F}|}e^{-\left(w_j^t+\eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t}-\lambda w_j^t\right)\right)}\right) \\ &\simeq w_j^t + \eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - 2\lambda w_j^t\right) - \eta\left(\frac{\epsilon\cdot|\mathcal{R}_j|}{|\mathcal{F}|}e^{-w_j^t}\left(1-\eta\left(\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} - \lambda w_j^t\right)\right)\right) \\ &= w_j^t - \eta\left(-\frac{|\mathcal{R}_j|}{|\mathcal{R}|}e^{-w_j^t} + \frac{\epsilon\cdot|\mathcal{R}_j|}{|\mathcal{F}|}e^{-w_j^t} + 2\lambda w_j^t\right) + \mathcal{O}(\eta^2). \end{aligned} \tag{4}$$

Eq. (4) shows that up to order $\mathcal{O}(\eta^2)$, the dynamics, as well as the convergent solution of the iterative descent-ascent algorithm are identical to the ones obtained from Eq. (2), up to a rescaling of the regularization parameter by a factor of 2, as in $\lambda_{\mathrm{DA}} = 2\lambda_{\mathrm{Iter-DA}}$.

G. Proof of Lemma 5

In this section we prove Lemma 5 under Assumption 1 and Assumption 2.

Lemma 5 (Closed Form)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ and w_j^{DA} be the jth coordinate of **any** local minima/maxima for the logistic regression problems defined in Eq. (1), then they admit the form:*

$$w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right), w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right), w_j^{DA} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right),$$

where $W(z)$ corresponds to the Lambert-W function, the solution to $z = W(z)e^{W(z)}$.

Proof. Let us start by restating the original problem as given in Eq. (2). For the sake of completeness.

$$\textbf{minimization } \mathcal{D}: \frac{1}{|\mathcal{D}|}\sum_{i=1}^{\mathcal{D}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

$$\textbf{minimization } \mathcal{R}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} - \frac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

We can get the local minima of these functions by using Fermat's theorem, therefore we have:

$$\textbf{minimization } \mathcal{D}: \frac{1}{|\mathcal{D}|}\sum_{i=1}^{\mathcal{D}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0$$

$$\textbf{minimization } \mathcal{R}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} - \frac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} -y_i\cdot x_i e^{-y_i\cdot\langle w,x_i\rangle} + \lambda w = 0$$

Solving the equations for coordinate j and using Assumption 1, we get:

$$\textbf{minimization } \mathcal{D}: w_j = \frac{1}{\lambda|\mathcal{D}|}(\sum_{i=1}^{S_j} y_i \cdot x_{i,j} e^{-y_i \cdot w_j \cdot x_{i,j}})$$

$$\textbf{minimization } \mathcal{R}: w_j = \frac{1}{\lambda|\mathcal{R}|}(\sum_{i=1}^{\mathcal{R}_j} y_i \cdot x_{i,j} e^{-y_i \cdot w_j \cdot x_{i,j}})$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: w_j = \frac{1}{\lambda|\mathcal{R}|}(\sum_{i=1}^{\mathcal{R}_j} y_i \cdot x_{i,j} e^{-y_i \cdot w_j \cdot x_{i,j}}) - \frac{1}{\lambda|\mathcal{F}|}(\sum_{i=1}^{\mathcal{F}_j} y_i \cdot x_{i,j} e^{-y_i \cdot w_j \cdot x_{i,j}})$$

Now we can utilize Assumption 2 and the fact that: $|\mathcal{F}_j| = \alpha \cdot |\mathcal{R}_j|$ to restate the previous equations in the form:

$$\textbf{minimization } \mathcal{D}: w_j = \frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|} e^{-w_j}$$

$$\textbf{minimization } \mathcal{R}: w_j = \frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|} e^{-w_j}$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: w_j = \frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|} e^{-w_j} - \frac{\alpha \cdot |\mathcal{R}_j|}{\lambda|\mathcal{F}|} e^{-w_j}$$

As explained in App. G.1 the Lambert function W provides the solution for equations of the previous form. Using this fact we get:

$$\textbf{minimization } \mathcal{D}: w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right)$$

$$\textbf{minimization } \mathcal{R}: w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: w_j^{\mathrm{DA}} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right)$$

This concludes the proof. □

G.1. The Lambert function W

In this section for the sake of exposition we briefly discuss the Lambert function W. Introduced by Johann Heinrich Lambert in 1758. In this work we are primarily interested in the property of the function that for any α, the solution of the equation:

$$x - \alpha \cdot e^{-x} = 0$$

is $x = \mathrm{W}(-a)$. As well as the monotonicity of the principal branch of the Lambert function.

H. Proof of Lemma 2

In this section of the appendix we provide the proof for Lemma 2, under Assumptions 1 and 2, we start by restating the Lemma below for the sake of exposition.

Lemma 2 (Divergence in Logistic Regression)**.** *Let $w_j^{(\mathcal{D})}$, $w_j^{(\mathcal{R})}$, and $w_j^{(DA)}$ be the jth coordinate of the convergence points for logistic regression on the original set $\mathcal{D}$, the retain set $\mathcal{R}$, and using the Descent-Ascent (DA) method, respectively. Then, for a specific range of α, we have: $\left(w_j^{(DA)} - w_j^{(\mathcal{D})}\right) \cdot \left(w_j^{(\mathcal{D})} - w_j^{(\mathcal{R})}\right) \geq 0$.*

Proof. To begin the proof let us restate the three minimization problems for logistic regression for the three cases, whose

respective solutions are $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}, w_j^{\text{DA}}$

$$\textbf{minimization } \mathcal{D}: \frac{1}{|\mathcal{D}|}\sum_{i=1}^{\mathcal{D}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

$$\textbf{minimization } \mathcal{R}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

$$\textbf{Descent } \mathcal{R} - \textbf{Ascent } \mathcal{F}: \frac{1}{|\mathcal{R}|}\sum_{i=1}^{\mathcal{R}} e^{-y_i\cdot\langle w,x_i\rangle} - \frac{1}{|\mathcal{F}|}\sum_{i=1}^{\mathcal{F}} e^{-y_i\cdot\langle w,x_i\rangle} + \frac{\lambda}{2}\|w\|_2^2$$

So the local minima and maxima of these equations can be characterized with the help of Lemma 5, the proof of which can be found in App. G, for the sake of completeness, let us restate the lemma here

Lemma 5 (Closed Form)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ and w_j^{DA} be the jth coordinate of **any** local minima/maxima for the logistic regression problems defined in Eq. (1), then they admit the form:*

$$w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right), w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right), w_j^{DA} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right),$$

where $W(z)$ corresponds to the Lambert-W function, the solution to $z = W(z)e^{W(z)}$.

Since $\alpha \geq 0$, we have that:

$$\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|} > 0 \text{ and } \frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|} > 0$$

The minimization for logistic regression over the original dataset $\mathcal{D}$ and the retrain dataset $\mathcal{R}$ both have a global minimum that is unique and corresponds to the solution of the principal branch of the Lambert function W_0, for that value.
For the Descent Ascent solution, since the input of the Lambert function is not necessarily positive, we have to separate our analysis to three cases:

1. The first case, where there is only one global minimum, meaning that the input x of the Lambert function is $x \geq 0$. Equivalently, we have $\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|} \geq 0$ which implies that $\alpha \leq \frac{|\mathcal{F}|}{|\mathcal{R}|}$
2. The second case, where we have a solution both for the primary and the secondary branch of the Lambert function, corresponding to a local maximum and minimum respectively meaning that you have that the input x of the Lambert function is $-1/e \leq x \leq 0$, equivalently solving for ϵ gives $|\mathcal{F}|/|\mathcal{R}| < \alpha \leq |\mathcal{F}|/|\mathcal{R}| + (\lambda|\mathcal{F}|)/(e|\mathcal{R}_j|)$
3. The third case, where there are no local minima, meaning that the input of the Lambert function x is $x < -1/e$, which implies that $\alpha > |\mathcal{F}|/|\mathcal{R}| + (\lambda|\mathcal{F}|)/(e|\mathcal{R}_j|)$

<u>Case 1:</u> In case 1 we have that $\alpha \leq \frac{|\mathcal{F}|}{|\mathcal{R}|}$, which implies that:

$$\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|} \leq \frac{(|\mathcal{R}|+|\mathcal{F}|)|\mathcal{R}_j|}{\lambda|\mathcal{R}||\mathcal{D}|} \leq \frac{|\mathcal{D}||\mathcal{R}_j|}{\lambda|\mathcal{R}||\mathcal{D}|} \leq \frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}$$

so since the principal branch W_0 of the Lambert function is increasing, we have that:

$$w_j^{\mathcal{D}} = \mathrm{W}_0\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right) \leq \mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) = w_j^{\mathcal{R}}$$

For this case, let us now assume that $\alpha \geq |\mathcal{F}|^2/\left(|\mathcal{R}|(|\mathcal{F}|+|\mathcal{D}|)\right)$, it is easy to verify that for such an α it holds that: $\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|} \geq \frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}$, so we have that:

$$w_j^{\text{DA}} = \mathrm{W}_0\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right) \leq \mathrm{W}_0\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right) = w_j^{\mathcal{D}}$$

So for **Case 1** we have that $w_j^{\text{DA}} \leq w_j^{\mathcal{D}} \leq w_j^{\mathcal{R}}$, which implies that $(w_j^{\text{DA}} - w_j^{\mathcal{D}}) \cdot (w_j^{\mathcal{D}} - w_j^{\mathcal{R}}) \geq 0$

This concludes the proof.

□

I. Proof of Lemma 6

In this section we provide the proof for Lemma 6 under Assumptions 1 and 2.

Lemma 6 (Distance Growth)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ the jth coordinate of the convergence point for the logistic regression problem for the original set $\mathcal{D}$ and the retain set $\mathcal{R}$ respectively. It holds that the distance $\Delta_{\mathcal{R},\mathcal{D}} = |w_j^{\mathcal{D}} - w_j^{\mathcal{R}}| \leq \left|\ln\left((1+\alpha)\frac{|\mathcal{R}|}{|\mathcal{D}|}\right)\right|$, for any value of $\lambda > 0$.*

Proof. We start from Lemma 5, which we restate below for the sake of exposition.

Lemma 5 (Closed Form)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ and w_j^{DA} be the jth coordinate of **any** local minima/maxima for the logistic regression problems defined in Eq. (1), then they admit the form:*

$$w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right), w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right), w_j^{DA} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right),$$

where $W(z)$ corresponds to the Lambert-W function, the solution to $z = W(z)e^{W(z)}$.

Since the input of the Lambert function for $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ is always positive these solutions correspond to the only minimum of the function for the minimization problem and additionally they are calculated from them principal branch of the Lambert function W_0. We start from the logarithmic connection of the Lambert function, which is that for any value of x it holds that:

$$\mathrm{W}(x) = \ln(x) - ln(\mathrm{W}(x))$$

So for $\alpha \geq \frac{|\mathcal{F}|}{|\mathcal{R}|}$,since W_0 is increasing we have that $w_j^{\mathcal{D}} \geq w_j^{\mathcal{R}}$ we have the following:

$$\begin{aligned}
\Delta_{\mathcal{R},\mathcal{D}} &= w_j^{\mathcal{D}} - w_j^{\mathcal{R}} \\
&= \mathrm{W}_0\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right) - \mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) \\
&= \mathrm{W}_0\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) - \mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) \quad \text{,where } \alpha = (1+\alpha)\frac{|\mathcal{R}|}{|\mathcal{D}|} \\
&= \ln\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) - \ln\left(\mathrm{W}_0\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)\right) - \ln\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) + \ln\left(\mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)\right) \\
&= \ln\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) - \ln\left(\mathrm{W}_0\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)\right) - \ln\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right) + \ln\left(\mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)\right) \\
&= \ln(\alpha) - \ln\left(\frac{\mathrm{W}_0\left(\alpha\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)}{\mathrm{W}_0\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right)}\right) \\
&\leq \ln(\alpha) \quad \text{, since the principal branch } \mathrm{W}_0 \text{ is increasing}
\end{aligned}$$

We can repeat the same proof procedure for $\alpha \leq \frac{|\mathcal{F}|}{|\mathcal{R}|}$, but instead we get $\Delta_{\mathcal{R},\mathcal{D}} \leq -\ln(\alpha)$. This concludes the proof □

J. Proof of Lemma 7

Lemma 7 (Distance Unlearning)**.** *Let $w_j^{\mathcal{R}}, w_j^{DA}$ the jth coordinate of the convergence point for the logistic regression problem for the retain set $\mathcal{R}$ and the Descent Ascent method respectively. It holds that for for $\alpha \geq |\mathcal{F}|/|\mathcal{R}|$ the distance $\Delta_{\mathcal{R},DA} = |w_j^{\mathcal{R}} - w_j^{DA}| \geq W_0\left(|\mathcal{R}_j|/(\lambda|\mathcal{R}|)\right)$*

Proof. We start from Lemma 5 which we restate below for the sake of exposition.

Lemma 5 (Closed Form)**.** *Let $w_j^{\mathcal{D}}, w_j^{\mathcal{R}}$ and w_j^{DA} be the jth coordinate of **any** local minima/maxima for the logistic regression problems defined in Eq. (1), then they admit the form:*

$$w_j^{\mathcal{D}} = \mathrm{W}\left(\frac{(1+\alpha)|\mathcal{R}_j|}{\lambda|\mathcal{D}|}\right), w_j^{\mathcal{R}} = \mathrm{W}\left(\frac{|\mathcal{R}_j|}{\lambda|\mathcal{R}|}\right), w_j^{DA} = \mathrm{W}\left(\frac{(1-\alpha|\mathcal{R}|/|\mathcal{F}|)|R_j|}{\lambda|R|}\right),$$

where $W(z)$ corresponds to the Lambert-W function, the solution to $z = W(z)e^{W(z)}$.

It is easy to notice that in the case where we have $\alpha = |\mathcal{F}|/|\mathcal{R}|$ $w_j^{\text{DA}} = 0$ which concludes this case. For the case where $\alpha > |\mathcal{F}|/|\mathcal{R}|$ we refer the reader to the proof of Lemma 2, where we show that $w_j^{\text{DA}} \to -\infty$ for any value of $\lambda > 0$ so the distance is infinite in this case. □

K. Logistic Regression 2 dimensions

In this section we will study the natural extension of the previous example, where we were studying the 1 dimensional case. In this case we assume that our samples are of the form:

$$s_1 = (1, \epsilon), \quad s_2 = (\epsilon, 1)$$

This gives the following equations for the optimality conditions for training on the full data set $\mathcal{D}$:

$$\begin{aligned} w_1 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(e^{-(w_1+w_2\epsilon)} + \alpha\epsilon e^{-(w_1\epsilon+w_2)}) \\ w_2 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(\epsilon e^{-(w_1+w_2\epsilon)} + \alpha e^{-(w_1\epsilon+w_2)}) \end{aligned}$$

For the retrain set $\mathcal{R}$ we have that:

$$\begin{aligned} w_1 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}e^{-(w_1+w_2\epsilon)} \\ w_2 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}\epsilon e^{-(w_1+w_2\epsilon)} \end{aligned}$$

For the Descent Ascent unlearning we have that:

$$\begin{aligned} w_1 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}e^{-(w_1+w_2\epsilon)} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha\epsilon e^{-(w_1\epsilon+w_2)} \\ w_2 &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}\epsilon e^{-(w_1+w_2\epsilon)} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha e^{-(w_1\epsilon+w_2)} \end{aligned}$$

We will now rewrite the above equations by setting $x = w_1 + w_2\epsilon$ and $y = w_1\epsilon + w_2$, this simplifies the equations and still allows us to make our claim that DA can only harm the model if there is a total ordering over the values of the solutions of the rewritten equations.

For the dataset $\mathcal{D}$ we have:

$$\begin{aligned} x^{\mathcal{D}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)e^{-x^{\mathcal{D}}} + 2\alpha\epsilon e^{-y^{\mathcal{D}}}) \\ y^{\mathcal{D}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon e^{-x^{\mathcal{D}}} + \alpha(1+\epsilon^2)e^{-y^{\mathcal{D}}}) \end{aligned}$$

For the retrain set $\mathcal{R}$, we have that:

$$
\begin{aligned}
x^{\mathcal{R}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)e^{-x^{\mathcal{R}}} \\
y^{\mathcal{R}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon e^{-x^{\mathcal{R}}}
\end{aligned}
$$

For the DA method we get the following equations:

$$
\begin{aligned}
x^{\text{DA}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)e^{-x^{\text{DA}}} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon e^{-y^{\text{DA}}} \\
y^{\text{DA}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon e^{-x^{\text{DA}}} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)e^{-y^{\text{DA}}}
\end{aligned}
$$

Before proceeding, let us point out that $y^{\text{DA}} \leq x^{\text{DA}}$, since $1+\epsilon^2 \geq 2\epsilon$, for the same reason, we get that $y^{\mathcal{R}} \leq x^{\mathcal{R}}$ and finally without loss of generality we will use that $y^{\mathcal{D}} \leq x^{\mathcal{D}}$. In Lemma 13 we give a short proof regarding the existence of such solutions.

Lemma 13. *For any $\alpha \leq 1$, we have that there exists a solution for the original dataset, such that $y^{\mathcal{D}} \leq x^{\mathcal{D}}$*

Proof. For $\alpha = 1$ we get that there exists a solution of the system such that $y^{\mathcal{D}} \leq x^{\mathcal{D}}$ by the symmetry of the system. For $\alpha \leq 1$. In order to demonstrate that there exists a solution for the system such that $y^{\mathcal{D}} \leq x^{\mathcal{D}}$ we will employ the nonlinear Gauss-Sidel method, which converges to a stationary point (minimum) for logistic regression. The proof goes as follows, we will initialize our algorithm in the solution for $\alpha = 1$ let it be x_0, y_0 and we know it holds that $x_0 \geq y_0$. We will follow the following update: (nonlinear Gauss-Sidel method starting from y)

$$
\begin{aligned}
y_{k+1} &\leftarrow 2b\epsilon e^{-x_k} + \mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_k}}\right) \\
x_{k+1} &\leftarrow 2b\alpha\epsilon e^{-y_k} + \mathrm{W}\left(b(1+\epsilon^2)e^{-2b\alpha\epsilon e^{-y_k}}\right)
\end{aligned}
$$

For y_1 we have that:

$$
\begin{aligned}
y_1 &= 2b\epsilon e^{-x_0} + \mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) \\
&= 2b\epsilon e^{-x_0} + \mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) - \mathrm{W}\left(b(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) + \mathrm{W}\left(b(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) \\
&= y_0 + \mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) - \mathrm{W}\left(b(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right)
\end{aligned}
$$

and since W is increasing we have that $\mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) - \mathrm{W}\left(b(1+\epsilon^2)e^{-2b\epsilon e^{-x_0}}\right) < 0$ implying that $y_1 < y_0$. Now let us define the function $f(x) = x + W\left(ce^{-x}\right)$ the function is increasing on x therefore since $y_1 < y_0$ we get that: $x_1 = f(2b\alpha\epsilon e^{-y_1}) > f(2b\alpha\epsilon e^{-y_0}) = x_0$. Let us proceed with an induction step, we assume that we have $x_k > x_{k-1}$ and $y_k < y_{k-1}$ for $k \geq 1$. We will show that $y_{k+1} < y_k$ which directly implies that $x_{k+1} = f(2b\alpha\epsilon e^{-y_{k+1}}) > f(2b\alpha\epsilon e^{-y_k}) = x_k$ completing the inductive step.

$$
\begin{aligned}
y_{k+1} &= 2b\epsilon e^{-x_k} + \mathrm{W}\left(b\alpha(1+\epsilon^2)e^{-2b\epsilon e^{-x_k}}\right) \\
&= f(2b\epsilon e^{-x_k}) < f(2b\epsilon e^{-x_{k-1}}) \\
&= y_k
\end{aligned}
$$

This concludes the inductive step and we therefore have that for all k $y_k \leq x_k$ for any α, as a result, since the method converges to the solution of the system there exists a solution which satisfies $y^{\mathcal{D}} \leq x^{\mathcal{D}}$. In the proof above we have that $b = ||\mathcal{R}_{i,j}|/\lambda|\mathcal{D}|$ □

K.1. Characterization of the solutions of the $2d$ Logistic regression

We start this section by giving an exact solution for the coordinates of the retrain problem.

Lemma 8. *The closed form solution for the stationary points for the retrain set is given as:*

$$x^{\mathcal{R}} = \mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{R}|}\right), \;\; y^{\mathcal{R}} = \frac{2\epsilon}{1+\epsilon^2}\mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{R}|}\right).$$

Proof. We have that:

$$x^{\mathcal{R}} \;=\; \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)e^{-x^{\mathcal{R}}} \rightarrow x^{\mathcal{R}} = \mathrm{W}\left(\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}\right)$$

So:

$$\begin{aligned} y^{\mathcal{R}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon e^{-x^{\mathcal{R}}} \\ &= \frac{2\epsilon}{1+\epsilon^2}\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}e^{-\mathrm{W}\left(\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}\right)} \\ &= \frac{2\epsilon}{1+\epsilon^2}\mathrm{W}\left(\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}\right) \end{aligned}$$

This concludes the proof. In the last equality we used the property of the Lambert function. □

For the other two problems it is not possible to provide exact solutions, as we did in the retrain one unfortunately, so we will provide upper and lower bounds for their values.

Lemma 9. *For the stationary points of the original set, one can derive the following ranges.*

$$\begin{aligned} \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)+2\alpha\epsilon)\right) &\leq x^{\mathcal{D}} \leq \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right) + \mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right), \\ \mathrm{W}\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right) &\leq y^{\mathcal{D}} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon+\alpha(1+\epsilon^2))\right). \end{aligned}$$

Proof. We have that

$$\begin{aligned} x^{\mathcal{D}} &= \frac{1}{\lambda|\mathcal{D}|}((1+\epsilon^2)e^{-x^{\mathcal{D}}} + 2\alpha\epsilon e^{-y^{\mathcal{D}}}) \\ y^{\mathcal{D}} &= \frac{1}{\lambda|\mathcal{D}|}(2\epsilon e^{-x^{\mathcal{D}}} + \alpha(1+\epsilon^2)e^{-y^{\mathcal{D}}}) \end{aligned}$$

As we discuss above we have that $y^{\mathcal{D}} \leq x^{\mathcal{D}} \Rightarrow e^{-y^{\mathcal{D}}} \geq e^{-x^{\mathcal{D}}}$, which implies that:

$$\begin{aligned} x^{\mathcal{D}} &\geq \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)e^{-x^{\mathcal{D}}} + 2\alpha\epsilon e^{-x^{\mathcal{D}}}) = \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)+2\alpha\epsilon)e^{-x^{\mathcal{D}}} \\ y^{\mathcal{D}} &\leq \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon e^{-y^{\mathcal{D}}} + \alpha(1+\epsilon^2)e^{-y^{\mathcal{D}}}) = \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon+\alpha(1+\epsilon)^2)e^{-y^{\mathcal{D}}} \end{aligned}$$

So from the inequalities above, we get that:

$$\begin{aligned} x^{\mathcal{D}} &\geq \mathrm{W}\left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)+2\alpha\epsilon)\right) \\ y^{\mathcal{D}} &\leq \mathrm{W}\left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon+\alpha(1+\epsilon^2))\right) \end{aligned}$$

Now we have an upper bound for $y^{\mathcal{D}}$ and a lower bound for $x^{\mathcal{D}}$. In order to provide a lower bound for $y^{\mathcal{D}}$ and an upper bound for $x^{\mathcal{D}}$. We should notice that $2\epsilon e^{-x^{\mathcal{D}}} \geq 0$, which gives:

$$\begin{aligned} y^{\mathcal{D}} &\geq \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\alpha(1+\epsilon^2)e^{-y^{\mathcal{D}}} \Rightarrow \\ y^{\mathcal{D}} &\geq \mathrm{W}\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right) \end{aligned}$$

This completes the bounds for $y^{\mathcal{D}}$, now in order to compute the upper bound for $x^{\mathcal{D}}$, we have that:

$$\begin{aligned} e^{-y^{\mathcal{D}}} &\leq e^{-\mathrm{W}\left(\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|/(\lambda|\mathcal{D}|)\right)} \Rightarrow \\ e^{-y^{\mathcal{D}}} &\leq \frac{\lambda|\mathcal{D}|}{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}e^{-\mathrm{W}\left(\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|/(\lambda|\mathcal{D}|)\right)} \Rightarrow \\ e^{-y^{\mathcal{D}}} &\leq \frac{\lambda|\mathcal{D}|}{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}W\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right) \end{aligned}$$

So we have that:

$$\begin{aligned} x^{\mathcal{D}} &\leq \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)e^{-x^{\mathcal{D}}} + 2\alpha\epsilon\frac{\lambda|\mathcal{D}|}{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}W\left(\frac{\alpha(1+\epsilon^2)}{\lambda|\mathcal{D}|}\right)) \Rightarrow \\ x^{\mathcal{D}} &\leq \frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}e^{-x^{\mathcal{D}}} + \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right)) \Rightarrow \\ x^{\mathcal{D}} &\leq \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right) + \mathrm{W}\left(\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}e^{-\frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right)}\right) \Rightarrow \\ x^{\mathcal{D}} &\leq \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right) + \mathrm{W}\left(\frac{(1+\epsilon^2)|\mathcal{R}_{i,j}|}{\lambda|\mathcal{D}|}\right) \end{aligned}$$

where the third inequality comes from the solution of the Lambert equation for the RHS of the inequality and the last one comes from the fact that the exponenent is non positive. This completes the proof. □

Lemma 10. *For the stationary points of the model trained by DA methods we can derive the following ranges.*

$$x^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2) - \frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon\right), \qquad y^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon - \frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)\right).$$

Proof. As stated earlier we have that $y^{\mathrm{DA}} \leq x^{\mathrm{DA}} \Rightarrow e^{-y^{\mathrm{DA}}} \geq e^{-x^{\mathrm{DA}}}$ and

$$\begin{aligned} x^{\mathrm{DA}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)e^{-x^{\mathrm{DA}}} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon e^{-y^{\mathrm{DA}}} \\ y^{\mathrm{DA}} &= \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon e^{-x^{\mathrm{DA}}} - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)e^{-y^{\mathrm{DA}}} \end{aligned}$$

So:

$$\begin{aligned} x^{\mathrm{DA}} &\leq \left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2) - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon\right)e^{-x^{\mathrm{DA}}} \\ y^{\mathrm{DA}} &\leq \left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)\right)e^{-y^{\mathrm{DA}}} \end{aligned}$$

So we get that:

$$\begin{aligned} x^{\mathrm{DA}} &\leq \mathrm{W}\left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2) - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon\right) \\ y^{\mathrm{DA}} &\leq \mathrm{W}\left(\frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon - \frac{|\mathcal{R}_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)\right) \end{aligned}$$

which concludes the proof. □

K.2. Derivation of the relevant size of the forget set

Lemma 11. *For* $\alpha \geq \alpha^{\mathcal{D}>DA} = \max\left\{\frac{1+\epsilon^2}{2\epsilon}\frac{|\mathcal{F}|^2}{|\mathcal{R}|(|\mathcal{D}|+|\mathcal{F}|)}, \frac{2\epsilon}{1+\epsilon^2}\frac{|\mathcal{F}||\mathcal{D}|}{|\mathcal{R}|(|\mathcal{D}|+|\mathcal{F}|)}\right\}$ *we have that* $x^{\mathcal{D}} \geq x^{DA}$ *and that* $y^{\mathcal{D}} \geq y^{DA}$.

Proof. We will start from Lemma 9 and Lemma 10, which we restate both below for the sake of exposition.

Lemma 9. *For the stationary points of the original set, one can derive the following ranges.*

$$\begin{aligned}
\mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}((1+\epsilon^2)+2\alpha\epsilon)\right) &\leq x^{\mathcal{D}} \leq \frac{2\epsilon}{1+\epsilon^2}W\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right)+\mathrm{W}\left(\frac{(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right),\\
\mathrm{W}\left(\frac{\alpha(1+\epsilon^2)|R_{i,j}|}{\lambda|\mathcal{D}|}\right) &\leq y^{\mathcal{D}} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{D}|}(2\epsilon+\alpha(1+\epsilon^2))\right).
\end{aligned}$$

Lemma 10. *For the stationary points of the model trained by DA methods we can derive the following ranges.*

$$x^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}(1+\epsilon^2)-\frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha 2\epsilon\right), \qquad y^{DA} \leq \mathrm{W}\left(\frac{|R_{i,j}|}{\lambda|\mathcal{R}|}2\epsilon-\frac{|R_{i,j}|}{\lambda|\mathcal{F}|}\alpha(1+\epsilon^2)\right).$$

We will require that the lower bounds provided for $x^{\mathcal{D}}, y^{\mathcal{D}}$ are bigger than the upper bounds provided for $x^{\mathrm{DA}}, y^{\mathrm{DA}}$, since the Lambert function W is monotone, we can just solve both inequalities for α, $x^{\mathcal{D}} \geq x^{\mathrm{DA}}$ and $y^{\mathcal{D}} \geq y^{\mathrm{DA}}$ and this concludes the proof. □

Finally we need to find the range of α for which it holds that $x^{\mathcal{R}} \geq x^{\mathcal{D}}$ and $y^{\mathcal{R}} \geq y^{\mathcal{D}}$, which is given in Lemma 4, which we restate next for the sake of exposition.

Lemma 4. *For* $\alpha \leq \alpha^{\mathcal{R}>\mathcal{D}} = \min\left\{\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}\right\}$ *we have that* $x^{\mathcal{R}} \geq x^{\mathcal{D}}$ *and that* $y^{\mathcal{R}} \geq y^{\mathcal{D}}$*, with* $\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}$.

Proof. We will use Lemma 9 and Lemma 8. Again similar to Lemma 11 we can solve for α and we get the expressions that solve the $x^{\mathcal{R}} > x^{\mathcal{D}}, y^{\mathcal{R}} > y^{\mathcal{D}}$ equations. Solving $x^{\mathcal{R}} = x^{\mathcal{D}}$

$$\alpha_x^{\mathcal{R}>\mathcal{D}} = \frac{D\lambda\left(W\left(\frac{\epsilon^2+1}{\lambda R}\right)-W\left(\frac{\epsilon^2+1}{D\lambda}\right)\right)\exp\left(\frac{(\epsilon^2+1)\left(W\left(\frac{\epsilon^2+1}{\lambda R}\right)-W\left(\frac{\epsilon^2+1}{D\lambda}\right)\right)}{2\epsilon}\right)}{2\epsilon}, \tag{5}$$

where for any $\alpha < \alpha_x^{\mathcal{R}>\mathcal{D}}$ there is a range of ϵ for which $x^{\mathcal{R}} > x^{\mathcal{D}}$.

Similarly, solving $y^{\mathcal{R}} = y^{\mathcal{D}}$

$$\alpha_y^{\mathcal{R}>\mathcal{D}} = \frac{2\epsilon\left(D\lambda e^{\frac{2\epsilon W\left(\frac{\epsilon^2+1}{\lambda R}\right)}{\epsilon^2+1}}W\left(\frac{\epsilon^2+1}{\lambda R}\right)-\epsilon^2-1\right)}{(\epsilon^2+1)^2}, \tag{6}$$

where for any $\alpha < \alpha_y^{\mathcal{R}>\mathcal{D}}$ there is a range of ϵ for which $y^{\mathcal{R}} > y^{\mathcal{D}}$.

The solution is therefore $\alpha \leq \min\left[\alpha_x^{\mathcal{R}>\mathcal{D}}, \alpha_y^{\mathcal{R}>\mathcal{D}}\right]$. □

L. Logistic Regression in 2D intuition

Let us consider nearly orthogonal data, such that all coordinates apart from two are orthogonal to each other. Namely, we choose the first two samples to be $x_1 = (1, \epsilon, 0, \ldots, 0)$ and $x_2 = (\epsilon, 1, 0, \ldots, 0)$, while the remaining $d-2$ points are orthogonal such that $x_a = e_a$ for $a = 3, \ldots, d$, where e_a are the unit vectors. We further assume that the two correlated samples x_1, x_2 share the same label $y_1 = y_2 = 1$. In this case, the unlearning problem decouples the first 2 dimensions from the rest, leaving a coupled set of equations for the weights along the first two directions w_1, w_2 for the original classification problem

$$w_1 = \frac{1}{\lambda|\mathcal{D}|}(e^{-(w_1+w_2\epsilon)}+\epsilon e^{-(w_1\epsilon+w_2)}), \quad w_2 = \frac{1}{\lambda|\mathcal{D}|}(\epsilon e^{-(w_1+w_2\epsilon)}+e^{-(w_1\epsilon+w_2)}), \tag{7}$$

which can be solved in the limit of $\epsilon \to 1^-$, as

$$w_1 = w_2 = \frac{1}{2} W\left(\frac{2(\epsilon+1)}{\lambda|\mathcal{D}|}\right). \tag{8}$$

The retrain problem has the minimum at

$$w_1 = \frac{1}{\lambda|\mathcal{R}|} e^{-(w_1+w_2\epsilon)}, \quad w_2 = \frac{1}{\lambda|\mathcal{R}|}\epsilon e^{-(w_1+w_2\epsilon)}, \tag{9}$$

and the DA is given by

$$w_1 = \frac{1}{\lambda|\mathcal{R}|}(e^{-(w_1+w_2\epsilon)} - \epsilon e^{-(w_1\epsilon+w_2)}), \quad w_2 = \frac{1}{\lambda|\mathcal{R}|}(\epsilon e^{-(w_1+w_2\epsilon)} - e^{-(w_1\epsilon+w_2)}). \tag{10}$$

Our goal is to study how far is the solution given by GDA from the one given by retraining. The retrained solution can be found analytically to be

$$w_1 = \frac{W\left(\frac{\epsilon^2+1}{|\mathcal{R}|\lambda}\right)}{\epsilon^2+1}, \quad w_2 = \frac{\epsilon W\left(\frac{\epsilon^2+1}{|\mathcal{R}|\lambda}\right)}{\epsilon^2+1}. \tag{11}$$

The GDA equations do not obtain a closed form solution, but they can be solved when assuming $\epsilon \to 1^-$, such that

$$w_1 = \frac{e^{-w_1-w_2}\left(w_1 - w_2 - 1\right)(\epsilon-1)}{\lambda|\mathcal{R}|}, \quad w_2 = \frac{e^{-w_1-w_2}\left(w_1 - w_2 + 1\right)(\epsilon-1)}{\lambda|\mathcal{R}|} \tag{12}$$

which are solved as

$$\begin{aligned} w_1 &= \frac{1}{4}\left(W\left(-\frac{8(\epsilon-1)^2}{|\mathcal{R}|^2\lambda^2}\right) - i\sqrt{2}\sqrt{W\left(-\frac{8(\epsilon-1)^2}{|\mathcal{R}|^2\lambda^2}\right)}\right), \\ w_2 &= \frac{1}{4}\left(W\left(-\frac{8(\epsilon-1)^2}{|\mathcal{R}|^2\lambda^2}\right) + i\sqrt{2}\sqrt{W\left(-\frac{8(\epsilon-1)^2}{|\mathcal{R}|^2\lambda^2}\right)}\right). \end{aligned} \tag{13}$$

It is sufficiently interesting to consider the sum of $w_1 + w_2$ compared to the retrained solution, and define the difference

$$\begin{aligned} \Delta = w_1^{\mathrm{DA}} + w_2^{\mathrm{DA}} - (w_1^{\mathrm{Re}} + w_2^{\mathrm{Re}}) &= \frac{1}{2}W\left(-\frac{8(\epsilon-1)^2}{|\mathcal{R}|^2\lambda^2}\right) - \frac{(1+\epsilon)W\left(\frac{\epsilon^2+1}{|\mathcal{R}|\lambda}\right)}{\epsilon^2+1} \\ &\underset{\epsilon\to 1^-}{=} -W\left(\frac{2}{|\mathcal{R}|\lambda}\right) \end{aligned} \tag{14}$$

M. Experimental details

Hyperparameters Following Georgiev et al. (2024) we pretrain ResNet-9 for 24 epochs using stochastic gradient descent (SGD) with an initial learning rate of 0.4, following a cyclic schedule that peaks at epoch 5. We employ a batch size of 512, momentum of 0.9, and a weight-decay coefficient of 5×10^{-4}.

We also adopt nine forget sets directly from Georgiev et al. (2024), which comprise both random subsets and semantically coherent subpopulations identified via principal-component analysis of the datamodel influence matrix. To construct them, an $n \times n$ datamodel matrix is formed by concatenating "train×train" datamodels (with $n = 50\,000$) by computing its top principal components (PCs) then we can define:

1. **Forget set 1**: 10 random samples.
2. **Forget set 2**: 100 random samples.
3. **Forget set 3**: 500 random samples.

4. **Forget set 4**: 10 samples with the highest projection onto the 1st PC.
5. **Forget set 5**: 100 samples with the highest projection onto the 1st PC.
6. **Forget set 6**: 250 samples with the highest and 250 samples with the lowest projection onto the 1st PC.
7. **Forget set 7**: 10 samples with the highest projection onto the 2nd PC.
8. **Forget set 8**: 100 samples with the highest projection onto the 2nd PC.
9. **Forget set 9**: 250 samples with the highest and 250 samples with the lowest projection onto the 2nd PC.

Most unlearning algorithms are highly sensitive to the choice of forget set and hyperparameters. Therefore we perform an extensive hyperparameter exploration, evaluating each baseline unlearning algorithm on each forget set. Our setting is again similar to Georgiev et al. (2024) but we consider a slightly larger hyperparameter grid for the employed methods and report results for all configurations rather than only the best-performing runs. More specifically, we evaluate over the Cartesian product of the following hyperparameter grids:

- **Gradient Ascent**: Optimized with SGD. Learning rates: $\{1 \times 10^{-5}, 5 \times 10^{-5}, 1 \times 10^{-4}, 5 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}, 5 \times 10^{-2}\}$; epochs: $\{1, 3, 5, 7, 10\}$.
- **Gradient Descent/Ascent**: Optimized with SGD. Learning rates: $\{5 \times 10^{-5}, 5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$; total epochs: $\{5, 7, 10\}$; ascent epochs: $\{3, 5\}$; forget batch size: $\{32, 64\}$.
- **SCRUB**: Optimized with SGD. Learning rates: $\{5 \times 10^{-5}, 5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}\}$; total epochs: $\{5, 7, 10\}$; ascent epochs: $\{3, 5\}$; forget batch size: $\{32, 64\}$.

We use a fixed batch size of 64 and train 100 models per configuration. For each run, we measure performance using the 95-th percentile of KLoM scores.

Statistical Significance Using $N = 100$ models to compute KLoM is computationally expensive although such expense comes at the gain of having low variance and results closely reproducing Georgiev et al. (2024). We find using lower values such as $N = 20$, $N = 50$ to produce large differences between margin distributions of pretrained and oracle models on the retain and validation sets (where KLoM should be low). More specifically, margin distributions become stable for all sets after $N = 80$. Reporting the 95-th percentile of KLoM scores follows the methodology established on Georgiev et al. (2024). Furthermore, reporting all runs instead of just the best one for each compute cost is more statistically transparent.

Compute resources All experiments were conducted on a server equipped with eight NVIDIA A100-SXM4 GPUs, each with 80 GB of GPU memory. A single unlearning configuration run was never split across different GPUs, many configurations were executed in parallel.

N. Additional Experiments

We provide additional analysis of the KLoM scores across various unlearning methods and forget sets. Fig. 6 presents the KLoM scores of Gradient Ascent, Gradient Descent/Ascent, and SCRUB. We observe that increasing the size of the forget set or including high-influence points significantly reduces the likelihood of achieving successful unlearning. Fig. 7 shows analogous results, but with KLoM scores computed over the retain set instead of the validation set. The patterns are nearly identical to those in Fig. 6. A pretrained model typically exhibits low KLoM scores on both validation and retain sets, with very similar magnitudes.

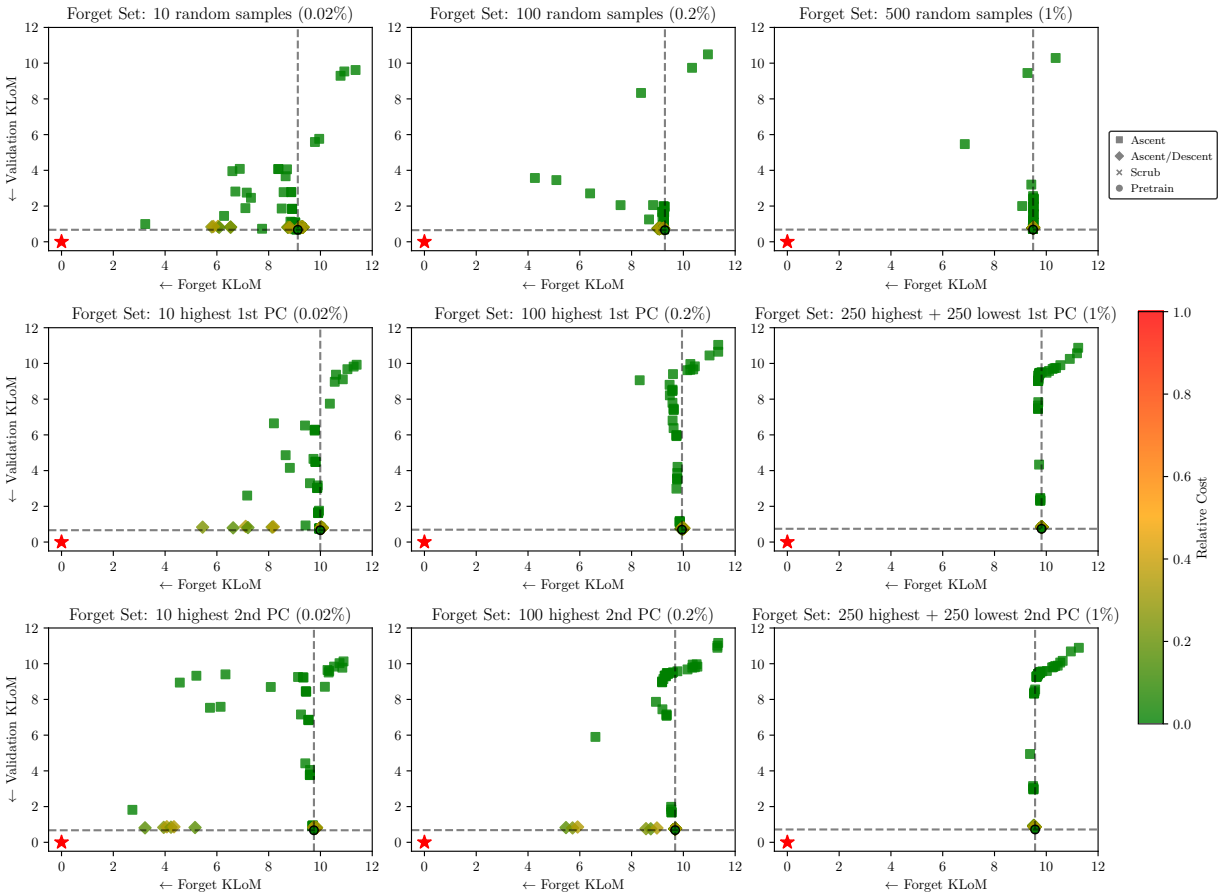

Figure 6: We present the KLoM scores of Gradient Ascent, Gradient Descent/Ascent and SCRUB when unlearning over each one of the forget sets (axes and points follow Fig. 1). We find an increase in forget set size and containing high influence points to strongly decrease the likelihood of any run achieving successful unlearning. For SCRUB we observe that runs remain close to the pretrained model in terms of KLoM scores under our experimental setup.

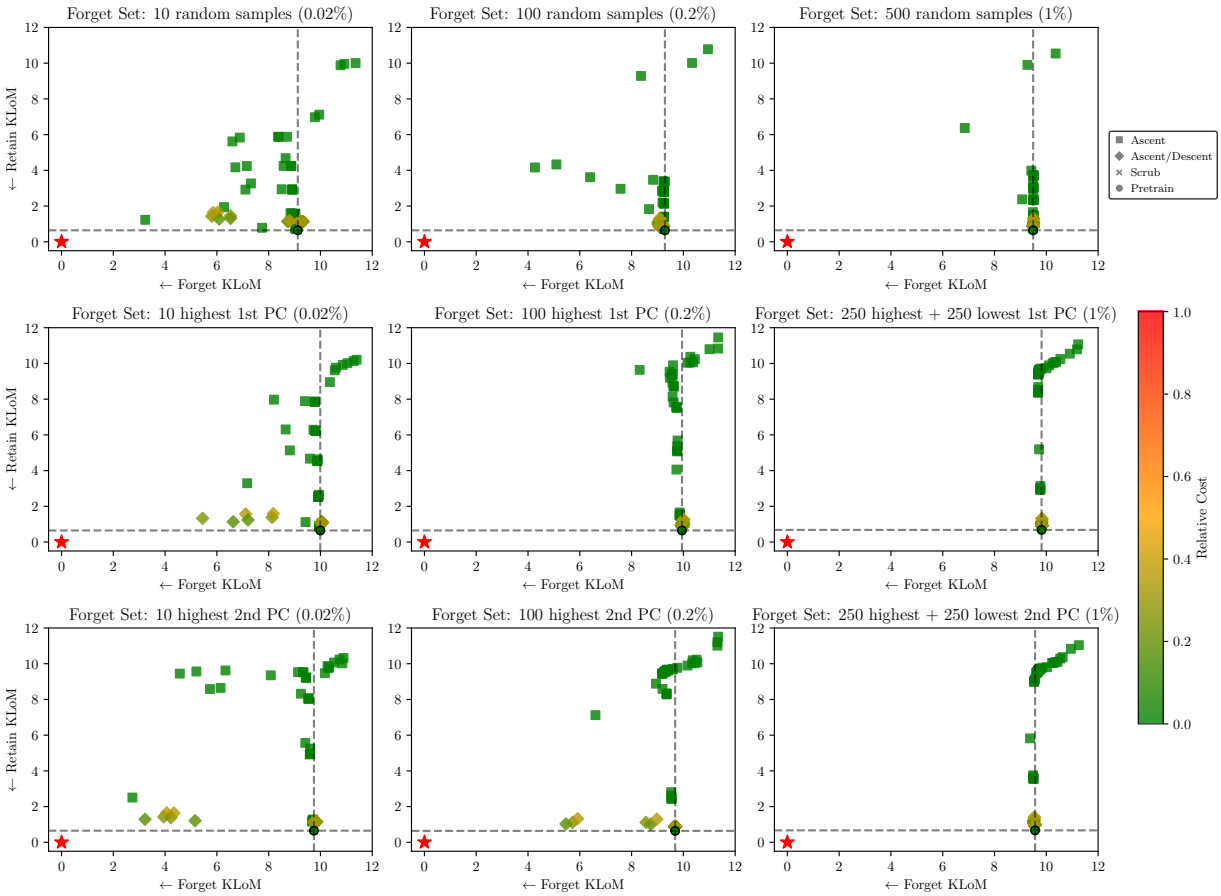

Figure 7: We present the KLoM scores of Gradient Ascent, Gradient Descent/Ascent and SCRUB when unlearning over each forget set. x-axis and points follow Fig. 1 and y-axis now displays the KLoM score in the retain set instead of the validation set. We observe very little difference when comparing with the results in Fig. 6. A pretrained model has low KLoM scores on both the validation and retain sets with very similar magnitudes. These findings are consistent with Georgiev et al. (2024).