ECAI 2023
K. Gal et al. (Eds.)
© 2023 The Authors.
This article is published online with Open Access by IOS Press and distributed under the terms of the Creative Commons Attribution Non-Commercial License 4.0 (CC BY-NC 4.0).
doi:10.3233/FAIA230407

Approximate Data Deletion in Generative Models

Zhifeng Kong[a;*] **and Scott Alfeld**[b;**]

[a]Computer Science and Engineering, UC San Diego, La Jolla, CA, USA
[b]Computer Science, Amherst College, Amherst, MA, USA
ORCiD ID: Scott Alfeld https://orcid.org/0000-0001-8446-4993

Abstract. Users have the right to have their data deleted by third-party learned systems, as codified by recent legislation such as the General Data Protection Regulation (GDPR) and the California Consumer Privacy Act (CCPA). Such data deletion can be achieved by full re-training, but this incurs a high computational cost for modern machine learning methods. To avoid this cost, many approximate deletion methods have been developed for supervised learning. Unsupervised learning, in contrast, remains largely an open problem when it comes to efficient approximate data deletion. In this paper, we introduce (1) an efficient method for approximate deletion in generative models, and (2) statistical tests for estimating whether training points have been deleted. We provide theoretical guarantees under various learner assumptions. We then empirically demonstrate our methods across a variety of generative methods.

1 Introduction

Machine learning has proved to be an increasingly powerful tool. With this power comes responsibility and there are growing concerns in academia, government, and the private sector about user privacy and responsible data management. Recent regulations (e.g., GDPR and CCPA) have introduced a *right to erasure* whereby a user may request that their data is deleted from a database. While deleting user data from database is straightforward, a savvy attacker might still be able to reverse-engineer the data by examining a machine learning model trained on it [1]. Re-training a model (after deleting the requested data) is computationally expensive, especially for modern deep learning methods [20]. This has motivated *machine unlearning* [7] where learned models are altered (in a computationally cheap way) to emulate the re-training process. In this paper we focus on machine unlearning for generative models, a class of unsupervised learning methods which learn the probability distribution from data.

Prior work in supervised learning proposed *approximate data deletion* to approximate the re-trained model without actually performing the re-training [17, 31, 40, 18]. While these methods have achieved great success, approximate data deletion for *un*supervised learning largely remains an open question. In this paper we present a density-ratio-based framework for approximate deletion in generative models. We present two novel contributions:

1. We propose a fast method for approximate data deletion for generative models.
2. We provide statistical tests to estimate whether training data have been deleted from a generative model given only sampling access to it.

For both contributions, we provide theoretical guarantees under a variety of learner assumptions. We also perform empirical investigations on real and synthetic datasets. In particular, our fast deletion algorithm is $> 10\times$ faster than re-training on real datasets.

The supervised and unsupervised settings have two major differences in the context of data deletion. The first is the definition – what does it mean to effectively delete training data? In the supervised setting, it is the classification function approximates the re-trained one, while in generative models it is to approximate the re-trained generative distribution. The other difference is the user's capability when evaluating data deletion. In the supervised setting, one can construct an input sample and query its predicted target. In contrast, a user can only draw samples from a generative model and then investigate the empirical distribution to evaluate the effectiveness of approximate data deletion.

In Section 2 we present our density-ratio-based framework and provide theoretical guarantee under various learner assumptions. We introduce practical algorithms for approximate deletion (our first primary contribution) in Section 3. We then study statistical tests with sampling access (our second primary contribution) in Section 4. We perform empirical investigations (Section 5) on real and synthetic data for both our fast deletion method and statistical test. We discuss related work in Section 6 and conclude with a discussion of future work in Section 7.

2 A Density Ratio Based Framework

Let p_* be a distribution over $\mathbb{R}^d$ and X be N i.i.d. samples from p_*. We consider a generative learning algorithm $\mathcal{A}$ which aims to model p_*. We denote the distribution $\mathcal{A}$ learns from X as $p_{\mathcal{A}(X)}$, and we refer to $\hat{p} = p_{\mathcal{A}(X)}$ as the pre-trained model. Let $X' \subset X$ be N' samples we would like to delete from $\hat{p}$, and $\hat{p}' = p_{\mathcal{A}(X \setminus X')}$ be the ground-truth re-trained model. A notation table is provided in Appendix A. In this paper, we present solutions to two problems:

1. Fast deletion: given $\hat{p}$, approximate $\hat{p}'$ more efficiently than full re-training.
2. Deletion test: assuming $q \in \{\hat{p}, \hat{p}'\}$, test whether $q = \hat{p}$ or $q = \hat{p}'$ by drawing samples.

* Corresponding Author. Email: z4kong@eng.ucsd.edu
** salfeld@amherst.edu

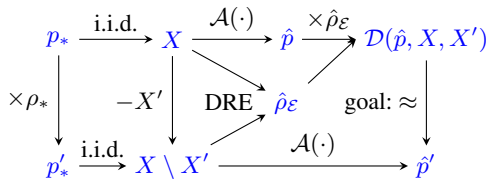


Figure 1: Our density-ratio-based framework for approximated data deletion for a generative learning algorithm $\mathcal{A}(\cdot)$. We train a DRE $\hat{\rho}_\mathcal{E}$ between X and $X \setminus X'$. We then multiply the DRE $\hat{\rho}_\mathcal{E}$ to the pre-trained model $\hat{p}$ to obtain $\mathcal{D}(\hat{p}, X, X')$. The goal is to let $\mathcal{D}(\hat{p}, X, X')$ approximate the re-trained model $\hat{p}'$. We model $X \setminus X'$ to be i.i.d. samples from p'_*, enabling theoretical guarantees.

Algorithm 1 Sampling from the approximated model

1: **Inputs:** $\hat{p}$, $\hat{\rho}_\mathcal{E}$.
2: **while** True **do**
3: Sample $y \sim \hat{p}$ and $u \sim \texttt{Uniform}([0, 1])$.
4: **if** $\hat{\rho}_\mathcal{E}(y) > B \cdot u$ **then**
5: **return** y
6: **end if**
7: **end while**

2.1 Framework

We present a density-ratio-based framework to perform fast (approximate) deletion and our deletion test. The density ratio between two distributions μ_1 and μ_2 on $\mathbb{R}^d$ is defined as $\rho(\mu_1, \mu_2) : \mathbb{R}^d \to \mathbb{R}^+, x \mapsto \mu_2(x)/\mu_1(x)$[1]. Let $\hat{\rho} = \rho(\hat{p}, \hat{p}')$ be the density ratio between pre-trained and re-trained models. In our proposed framework, we learn a density ratio estimator (DRE) $\hat{\rho}_\mathcal{E} = \text{DRE}(X, X \setminus X')$ between X and $X \setminus X'$ to approximate $\hat{\rho}$. Then, to perform fast deletion we define the approximated model $\mathcal{D}(\hat{p}, X, X') : \mathbb{R}^d \to \mathbb{R}^+, x \mapsto \hat{\rho}_\mathcal{E}(x) \cdot \hat{p}(x)$, which we abbreviate as $\hat{\rho}_\mathcal{E} \cdot \hat{p}$ for conciseness.

Core to both our method of fast deletion and our deletion test is our DRE based framework (summarized in Fig. 1). We model $X \setminus X'$ to be a set of i.i.d. samples from some distribution we denote as p'_*, and define $\rho_* = \rho(p_*, p'_*)$. We assume $\|\rho_*\|_\infty \leq \infty$. Intuitively, deleting some samples from p_* will only increase likelihood of regions far from these samples by at most a constant factor, and reduce likelihood of regions around these samples. We also assume $N' \ll N$ (only a small fraction of training samples are to be deleted). Intuitively, the pre-trained and re-trained models are likely similar. This allows us to provide approximation bounds for consistent learning algorithms $\mathcal{A}$. In Section 2.2 we derive such bounds for various forms of consistency.

In the supervised setting, approximate deletion can be done by altering the pre-trained model to be closer to the (never computed) re-trained model. In contrast, we alter the process of sampling from the unsupervised pre-trained model to simulate sampling from the re-trained model. Drawing samples from the approximated model is done in two steps: first draw samples from $\hat{p}$, and then perform rejection sampling according to $\hat{\rho}_\mathcal{E}$. Note that this procedure requires there exists a known constant $B \geq \|\hat{\rho}_\mathcal{E}\|_\infty$, which we discuss further in Section 2.3. We present this procedure in Alg. 1.

2.2 Approximation under Consistency

A learning algorithm $\mathcal{A}$ is said to be *consistent* if $p_{\mathcal{A}(X)}$ converges to p_* as $N \to \infty$ [45], where each specific type of convergence leads to a specific definition of consistency. If $\mathcal{A}$ is consistent, then we have $\hat{p} \approx p_*$ and $\hat{p}' \approx p'_*$ for large N. In this section, we derive DREs for two kinds of consistency to achieve approximated deletion: $\hat{\rho}_\mathcal{E}$ such that the approximated model $\mathcal{D}(\hat{p}, X, X') := \hat{\rho}_\mathcal{E} \cdot \hat{p} \approx \hat{p}'$.

In **Def.** 1, we introduce ratio consistency, which bounds the density ratio between true and learned distributions. We show in **Thm.** 2 that approximation in L_1 distance can be achieved in this case.

Definition 1 (Ratio Consistent (RC)). *$\mathcal{A}$ is (c_N, δ_N)-RC if for any density μ, with probability at least $1 - \delta_N$, it holds that $\|\log \rho(p_{\mathcal{A}(Z)}, \mu)\|_\infty \leq \log c_N$, where Z contains N i.i.d. samples from μ, and $c_N \to 1$, $\delta_N \to 0$ as $N \to \infty$.*

Theorem 2 (Approximation under RC). *If $\mathcal{A}$ is (c_N, δ_N)-RC, then there exists a DRE $\hat{\rho}_\mathcal{E}$ such that with probability at least $1 - 2(\delta_N + \delta_{N-N'})$, it holds that $\|\hat{\rho}_\mathcal{E} \cdot \hat{p} - \hat{p}'\|_1 \leq 4(c_N + c_{N-N'} - 2)$.*

We then look at a more practical definition of consistency in **Def.** 3, which bounds the total variation distance (half of L_1 distance) between true and learned distributions. We show in **Thm.** 4 that approximation in expectation is achieved in this case.

Definition 3 (Total Variation Consistent (TVC)). *$\mathcal{A}$ is (ϵ_N, δ_N)-TVC if for any density μ, with probability at least $1 - \delta_N$, it holds that $\|p_{\mathcal{A}(Z)} - \mu\|_1 \leq \epsilon_N$, where Z contains N i.i.d. samples from μ, and $\epsilon_N \to 0$, $\delta_N \to 0$ as $N \to \infty$.*

Theorem 4 (Approximation under TVC). *Define $\|h\|_{1,\mu} = \int_x \mu(x)|h(x)|dx$. If $\mathcal{A}$ is (ϵ_N, δ_N)-TVC, then there exists a DRE $\hat{\rho}_\mathcal{E}$ such that with probability at least $1 - 2(\delta_N + \delta_{N-N'})$, it holds that $\|\hat{\rho}_\mathcal{E} \cdot \hat{p} - \hat{p}'\|_{1,\hat{p}} \leq 2(\epsilon_{N-N'} + \|\rho_*\|_\infty \epsilon_N)$.*

We prove these theorems by construction. Full proofs are provided in Appendix B.1. For each, the high level idea is to choose a fixed consistent algorithm $\mathcal{A}_0$, and define $\hat{\rho}_\mathcal{E}(Z_1, Z_2) = \rho(p_{\mathcal{A}_0(Z_1)}, p_{\mathcal{A}_0(Z_2)})$. This yields $\hat{\rho}_\mathcal{E}(X, X \setminus X') \approx \rho_* \approx \hat{\rho}$ and therefore $\mathcal{D}(\hat{p}, X, X') = \hat{\rho}_\mathcal{E} \cdot \hat{p} \approx \hat{p}'$. We summarize the results in Table 1.

2.3 Practicability under Stability

In practice, we need $\|\hat{\rho}_\mathcal{E}\|_\infty$ to be finite in order to perform rejection sampling in Alg. 1 (see Line 3). Under this constraint, to satisfy $\hat{\rho}_\mathcal{E} \approx \hat{\rho}$, we need $\|\hat{\rho}\|_\infty$ to be finite as a prerequisite. In this section, we study several stability conditions of the learning algorithm $\mathcal{A}$ that guarantee $\|\hat{\rho}\|_\infty$ to be finite.

We organize these stability conditions in the order from *strong* to *weak*. We first discuss several strong, classic stability conditions that guarantee $\|\hat{\rho}\|_\infty$ to be small (**Def.** 5 – 7, **Thm.** 8). To state our definitions, let Z ($\hat{Z}$) be any training (test) set and z ($\hat{z}$) be any sample in Z ($\hat{Z}$).

Definition 5 (Differentially Private (DP) [11]). *$\mathcal{A}$ is ϵ-DP if $|\log p_{\mathcal{A}(Z\setminus\{z\})}(\hat{Z}) - \log p_{\mathcal{A}(Z)}(\hat{Z})| \leq \epsilon$.*

Definition 6 (Uniformly Stable (US) [5]). *$\mathcal{A}$ is ϵ-US if $|\log p_{\mathcal{A}(Z\setminus\{z\})}(\hat{z}) - \log p_{\mathcal{A}(Z)}(\hat{z})| \leq \epsilon$.*

Definition 7 (Lower Bounded in Likelihood Influence (LBLI) [25]). *$\mathcal{A}$ is ϵ-LBLI if $p_{\mathcal{A}(Z\setminus\{z\})}(\hat{z}) \leq e^\epsilon p_{\mathcal{A}(Z)}(\hat{z})$.*

We discuss relationship among DP, US and LBLI algorithms below. Note that ϵ-DP implies ϵ-US and ϵ-US implies ϵ-LBLI. If $\mathcal{A}$ is ϵ-DP or ϵ-US, the re-trained model satisfies $\hat{p}' \approx \hat{p}$, and there is no need to perform deletion. If $\mathcal{A}$ is ϵ-LBLI but not ϵ-US, then there exists

[1] We choose this order for cleaner theory.

a sample $\hat{z}$ such that $\hat{p}'(\hat{z}) \ll \hat{p}(\hat{z})$. Intuitively, in non-parametric methods, $\hat{z}$ can be samples near X'. ϵ-LBLI can be achieved under some regulatory assumptions on the loss function and the Hessian matrix with respect to parameters [13, 14, 2]. Then, we have the following result.

Theorem 8. *If $\mathcal{A}$ is ϵ-DP, ϵ-US, or ϵ-LBLI, then* $\log \|\hat{\rho}\|_\infty \leq N'\epsilon$.

Next, we move on to weaker stability assumptions to the learner. We introduce ratio stability, a concept crafted for our framework (**Def.** 9), which bounds the difference between log density ratios of true and learned distributions. We discuss its connection with ratio consistency (**Thm.** 10), and bound the difference between $\|\hat{\rho}\|_\infty$ and $\|\rho_*\|_\infty$ (**Thm.** 11). Finally, we discuss a special type of error stability [5] (**Def.** 12) and show a concentration bound on $\hat{\rho}$ (**Thm.** 13).

Definition 9 (Ratio Stable (RS))**.** *$\mathcal{A}$ is (ϵ, δ)-RS if for any densities μ_1, μ_2 such that $\sup_x \mu_2(x)/\mu_1(x) < \infty$, with probability at least $1-\delta$, when i.i.d. samples $Z_i \sim \mu_i$ satisfy $|Z_1| = |Z_2| + 1$, it holds that $\| \log \rho(\mu_1, p_{\mathcal{A}(Z_1)}) - \log \rho(\mu_2, p_{\mathcal{A}(Z_2)})\|_\infty \leq \epsilon$.*

Theorem 10. *If $\mathcal{A}$ is (c_N, δ_N)-RC, then $\mathcal{A}$ is $(2\log c_N, 2\delta_N)$-RS.*

Theorem 11. *If $\mathcal{A}$ is (ϵ, δ)-RS, then with probability at least $1 - N'\delta$, it holds that* $\log \|\hat{\rho}\|_\infty \leq N'\epsilon + \log \|\rho_*\|_\infty$.

Definition 12 (Error Stable (ES) [5])**.** *$\mathcal{A}$ is (ϵ, k)-ES if* $|\mathbb{E}_{\hat{z}\sim p_{\mathcal{A}(Z)}} \left[\log \left(p_{\mathcal{A}(Z\setminus\{z\})}(\hat{z})/p_{\mathcal{A}(Z)}(\hat{z})\right)\right]^k| < \epsilon$.

Theorem 13. *Let $N' = 1$. If $\mathcal{A}$ is $(\epsilon, 2)$-ES, then with probability at least $1-\delta$, it holds that* $\log \hat{\rho}(x) \leq \sqrt{\epsilon(1-\delta)/\delta}$ *for $x \sim \hat{p}$.*

We prove these theorems by induction and statistical inequalities. See Appendix B.2 for proofs. We summarize the results in Table 2.

Table 1: High-level summary of approximation results between the approximated model $\mathcal{D}(\hat{p}, X, X')$ and the re-trained model $\hat{p}'$ under different consistency assumptions to the learner $\mathcal{A}$.

Assumption	Approximation Result
RC (**Def.** 1)	in $\|\cdot\|_1$ with high probability
TVC (**Def.** 3)	in $\|\cdot\|_{1,\hat{p}}$ with high probability

Table 2: High-level summary of bounds on $\log \|\hat{\rho}\|_\infty$ under different stability assumptions to the learner $\mathcal{A}$.

Assumption	Bound on $\log \|\hat{\rho}\|_\infty$
DP (**Def.** 5)	$\mathcal{O}(\epsilon)$
US (**Def.** 6)	$\mathcal{O}(\epsilon)$
LBLI (**Def.** 7)	$\mathcal{O}(\epsilon)$
RS (**Def.** 9)	const $+ \mathcal{O}(\epsilon)$ with high probability.
ES (**Def.** 12)	$\mathcal{O}\left(\sqrt{\epsilon/\delta}\right)$ with probability $1-\delta$.

3 Density Ratio Estimators for Fast Data Deletion

A key step in the proposed framework is to train a density ratio estimator (DRE) $\hat{\rho}_\mathcal{E}$ between X and $X \setminus X'$. There is a rich literature of DRE techniques [43, 34, 30, 22, 37, 21, 9, 10]. All of these methods are designed for settings with little or no prior information about the data. We leverage the strong prior information that one set ($X \setminus X'$) is a strict subset of the other (X) to design more focused DRE methods for our data deletion setting. In Section 3.1, we derive a simple DRE based on probabilistic classification, and compare it with standard methods [43]. In Section 3.2, we use variational divergence minimization [34] to train a DRE that can handle high dimensional real-world datasets.

3.1 Probabilistic Classification

We derive a simple DRE based on probabilistic classification [43]. Let f be a classifier on $\{X \setminus X', X'\}$, where $f(x) = \text{Prob}(x \in X')$. Let $\texttt{nu}$ be the event that $X \setminus X'$ is used to train the model, and $\texttt{de}$ be the event that X is used to train the model. We apply the Bayes rule as follows [2]:

$$\begin{aligned}\hat{\rho}_\mathcal{E}(x) &= \frac{\text{Prob}(x|\texttt{nu})}{\text{Prob}(x|\texttt{de})} = \frac{\text{Prob}(\texttt{nu}|x)/\text{Prob}(\texttt{nu})}{\text{Prob}(\texttt{de}|x)/\text{Prob}(\texttt{de})} \\ &= \frac{N}{N-N'} \cdot \frac{\frac{1}{2}\text{Prob}(x \in X \setminus X')}{\text{Prob}(x \in X') + \frac{1}{2}\text{Prob}(x \in X \setminus X')} \\ &= \frac{N}{N-N'} \cdot \frac{\frac{1}{2}(1-f(x))}{f(x) + \frac{1}{2}(1-f(x))} \\ &= \frac{N}{N-N'} \cdot \frac{1-f(x)}{1+f(x)}.\end{aligned}$$

As an example, consider Kernel Density Estimation (KDE) [38, 35], a class of consistent algorithms which learn an explicit probability density.

Example 14 (KDE)**.** *Let $\mathcal{A}$ be KDE with Gaussian kernel function $K_\sigma(x) = \mathcal{N}(x; 0, \sigma^2 I)$. Then, The following classifier f exactly recovers $\hat{\rho}_\mathcal{E} = \hat{\rho}$:*

$$f(x) = \frac{\sum_{i=1}^{N'} K_\sigma(x - x_i)}{\left(\sum_{i=1}^{N'} + 2\sum_{i=N'+1}^{N}\right) K_\sigma(x - x_i)}. \tag{1}$$

Example 14 indicates that we need to up-weight samples in $X \setminus X'$ in the classifier, in addition to the DRE in the general setting derived by [43]. This observation is universal as $1 - f(x)$ is shared by both cases ($x \in X$ and $x \in X \setminus X'$) when we compute DRE.

3.2 Variational Divergence Minimization

Note that KDE and classification-based DRE are especially amenable to our methods but may not be able to deal with complicated, high-dimensional datasets [10]. Now, we consider the learner to be a Generative Adversarial Network (GAN) [15], a class of powerful implicit deep generative models. For these models, we derive a DRE based on variational divergence minimization (VDM) [34]. Because neural networks can have large capacity and VDM is designed to distinguish distributions, VDM-based DRE is more applicable with complicated data such as images compared to classification-based DRE. We begin with the definition of ϕ-divergence below.

Definition 15 ([27])**.** *Let $\phi : [0, \infty) \to \mathbb{R}$ be a strictly convex function such that $\phi(x)$ is finite for $x > 0$, $\phi(1) = 0$ and $\phi(0) = \lim_{x\to 0^+} \phi(x)$. The ϕ-divergence between distributions μ and ν is defined as $D_\phi(\mu\|\nu) = \int_x \nu(x)\phi\left[\mu(x)/\nu(x)\right] dx$.*

$D_\phi(p'_*||p_*)$ satisfies the following variational bound [33]:

$$D_\phi(p'_*||p_*) \geq \sup_T \left(\mathbb{E}_{x\sim p'_*} T(x) - \mathbb{E}_{x\sim p_*} \phi^*(T(x))\right), \tag{2}$$

where ϕ^* is the conjugate function of ϕ defined as $\phi^*(t) := \sup_u(ut - \phi(u))$. The optimal T is $T(x) = \frac{d}{dt}\phi(p'_*(x)/p_*(x)) = \frac{d}{dt}\phi(\rho_*(x))$, and in this case (2) achieves equality. Then, VDM is optimizing the right-hand-side of (2), usually via a neural network. Once the optimal T is obtained, we can solve $\hat{\rho}_\mathcal{E} = (\frac{d}{dt}\phi)^{-1}(T)$.

[2] The notations $\texttt{nu}$ and $\texttt{de}$ follow [43].

To perform the actual training in practice, we optimize the empirical version of the lower bound (2) based on the i.i.d. assumptions on X and $X \setminus X'$.

$$T_\phi = \arg\max_T \mathbb{E}_{x\sim X\setminus X'} T(x) - \mathbb{E}_{x\sim X}\phi^*(T(x)), \quad (3)$$

We provide specific algorithms to train DRE for two ϕ-divergences below. In both examples, T is a neural network.

Example 16 (Jensen-Shannon)**.** *Let D_ϕ be Jason-Shannon divergence. With an additional $\log(\cdot)$ term, we recover the discriminator loss in GAN [15]:*

$$T_\phi = \arg\max_T \mathbb{E}_{x\sim X\setminus X'} \log T(x) + \mathbb{E}_{x\sim X} \log(1 - T(x)). \quad (4)$$

In this case, the estimated density ratio is $\hat{\rho}_\phi = T_\phi/(1 - T_\phi)$.

Example 17 (Kullback–Leibler)**.** *Let D_ϕ be KL divergence. Then, we recover the discriminator loss in KL-GAN [28]:*

$$T_\phi = \arg\max_T \mathbb{E}_{x\sim X\setminus X'} T(x) - \mathbb{E}_{x\sim X} e^{T(x)}. \quad (5)$$

In this case, the estimated density ratio is $\hat{\rho}_\phi = \exp(T_\phi - 1)$.

Note that given enough capacity and data, we have $\hat{\rho}_\phi \approx \rho_*$ rather than $\hat{\rho}$, which may cause some bias. This bias can be alleviated when the learner $\mathcal{A}$ is consistent and expressive enough, such as in the case of GANs [29]. We find KL divergence works well in practice.

4 Statistical Tests for Data Deletion

Our second main contribution is our statistical deletion tests to distinguish whether a generative model has has particular points deleted. Formally, we assume sample access to a distribution q, which is either the pre-trained model $\hat{p}$ or the re-trained model $\hat{p}'$. We consider the following hypothesis test: $H_0 : q = \hat{p}$; $H_1 : q = \hat{p}'$.[3] Several statistics for this test (not in the data deletion setting) have been proposed, including likelihood ratio (LR) [32], Ali-Silvey-Csiszár (ASC) statistics [19], and maximum mean discrepancy (MMD) [16]. In this section, we adapt LR and ASC to the data deletion setting, and discuss MMD in Appendix C.3. In practice, we may not know $\hat{p}'$, so we use $H_1' : q = \mathcal{D}(\hat{p}, X, X')$ to approximate H_1. We present theory on the approximation between H_1 and H_1' when these statistics are used, thus providing an efficient way to test H_0 vs H_1 without re-training.[4]

4.1 Likelihood Ratio

A common goodness-of-fit method is the likelihood ratio test. In terms of having the smallest type-2 error, the likelihood ratio test is the most powerful of statistical tests [32] and is performed as follows. Given m samples $Y \sim q$, the likelihood ratio statistic is defined as

$$\mathrm{LR}(Y, \hat{p}, \hat{p}') = \frac{1}{m}\sum_{y\in Y} \log\frac{\hat{p}'(y)}{\hat{p}(y)} = \frac{1}{m}\sum_{y\in Y}\log\hat{\rho}(y).$$

As it is solely determined by Y and $\hat{\rho}$, we abbreviate it as $\mathrm{LR}(Y, \hat{\rho})$. When we use H_1' to approximate H_1 in practice, we compute $\mathrm{LR}(Y, \hat{\rho}_\mathcal{E})$. By **Thm.** 18, it approximates $\mathrm{LR}(Y, \hat{\rho})$ with high probability under RC (**Def.** 1), and in **Thm.** 19, we show approximation when $\hat{\rho}_\mathcal{E}$ is close to $\hat{\rho}$. Statistical properties of likelihood ratio and proofs to these theorems are in Appendix C.1.

[3] This is different from two-sample tests, where H_1 is $q \neq \hat{p}$, and we do not have knowledge of $\hat{p}'$.

[4] It is unclear how to test H_0 vs H_1 even with re-training if $\mathcal{A}$ does not yield explicit likelihood (e.g., GAN).

Theorem 18. *If $\mathcal{A}$ is (c_N, δ_N)-RC, then there exists a $\hat{\rho}_\mathcal{E}$ such that with probability at least $1-2(\delta_N+\delta_{N-N'})$, it holds that $|\mathrm{LR}(Y, \hat{\rho}) - \mathrm{LR}(Y, \hat{\rho}_\mathcal{E})| \le 2(\log c_N + \log c_{N-N'})$.*

Theorem 19. *(1) If $\|\log\hat{\rho} - \log\hat{\rho}_\mathcal{E}\|_\infty \le \epsilon$, then $|\mathrm{LR}(Y, \hat{\rho}) - \mathrm{LR}(Y, \hat{\rho}_\mathcal{E})| \le \epsilon$.*
(2) If $\max(\|\log\hat{\rho} - \log\hat{\rho}_\mathcal{E}\|_{1,\hat{p}}, \|\log\hat{\rho} - \log\hat{\rho}_\mathcal{E}\|_{1,\hat{p}'}) \le \epsilon$, then with probability at least $1-\delta$, it holds that $|\mathrm{LR}(Y, \hat{\rho}) - \mathrm{LR}(Y, \hat{\rho}_\mathcal{E})| \le \epsilon/\delta$.

4.2 ASC Statistics

ASC statistics are used to estimate the ϕ-divergence (**Def.** 15) [19]. Because a broad family of ϕ functions can be used, these statistics include a wide range of statistics. Drawing m samples $Y \sim q$ and another m samples $\hat{Y}$ from $\hat{p}$, the ASC statistic is defined as

$$\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) = \frac{1}{m}\Big[\sum_{y\in\hat{Y}} + \sum_{y\in Y}\Big]\frac{\phi(\hat{\rho}(y))}{1+\hat{\rho}(y)}.$$

When we use H_1' to approximate H_1 in practice, we compute $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}_\mathcal{E})$. In **Thm.** 20, we show it approximates $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho})$ when $\hat{\rho}_\mathcal{E}$ is close to $\hat{\rho}$.

Theorem 20. *If $\max(\|\psi(\hat{\rho}) - \psi(\hat{\rho}_\mathcal{E})\|_{1,\hat{p}}, \|\psi(\hat{\rho}) - \psi(\hat{\rho}_\mathcal{E})\|_{1,\hat{p}'}) \le \epsilon$ where $\psi(t) = \phi(t)/(1+t)$, then with probability at least $1-\delta$, it holds that $|\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) - \hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}_\mathcal{E})| \le 2\epsilon/\delta$.*

Statistical properties of ASC statistics and our proof of the above theorem are in Appendix C.2.

5 Experiments

Empirically, we address the following questions. **1) DRE Approximations**: do the methods in Section 3 produce ratios $\hat{\rho}_\mathcal{E}$ that approximate the target ratio $\hat{\rho}$? **2) Fast Deletion**: is $\mathcal{D}(\hat{p}, X, X') = \hat{\rho}_\mathcal{E}\cdot\hat{p}$ indistinguishable from the re-trained model $\hat{p}'$? And **3) Hypothesis Test**: do tests in Section 4 distinguish samples from pre-trained and re-trained models?

We first survey these questions in experiments on two-dimensional synthetic datasets. We then look at GANs trained on MNIST [26] and Fashion-MNIST [46].

5.1 Classification-based DRE for KDE on Synthetic Datasets

Experiment setup. We generate two synthetic distributions (p_*) over $\mathbb{R}^2$ based on mixtures: a mixture of 8 Gaussian distributions (MoG-8) (Fig. 2a), and a checkerboard distribution with 8 squares on a 4×4 checkerboard (CKB-8) (Appendix D.2). We define p_*' to be a weighted mixture version of p_*: four re-weighted clusters have weight $\lambda \in (0, 1)$ (for MoG-8, they are the clusters at 3, 6, 9, and 12 o'clock), and the other four have weight 1 (see Fig. 2b). We draw $N = 400$ samples from p_* to form X, and randomly reject $1-\lambda$ fraction of samples in re-weighted clusters to form the deletion set X' (see Fig. 2f). We run KDE using a Gaussian kernel and $\sigma_\mathcal{A} = 0.1$ to obtain pre-trained models in Fig. 2c, re-trained models in Fig. 2d, and their ratio $\hat{\rho}$ in Fig. 2e. We use KDE because its density is explicit and thus we are able to compute the exact likelihood ratio to examine the effectiveness of our DRE-based framework.

Method and results. We use the classification-based DRE described in Section 3.1. We up-weigh $X \setminus X'$ when training the classifiers according to Example 14. We consider two types of non-parametric classifiers: kernel-based classifiers (KBC) defined in (1) with potentially different $\sigma = \sigma_{\mathcal{C}} \neq \sigma_{\mathcal{A}}$, and k-nearest-neighbour classifiers (kNN) defined as the fraction of positive votes in k nearest neighbours.[5] For each classifier, we draw four sets of i.i.d. samples (each of size m):

1. $\hat{Y} \sim \hat{p}$ (pre-trained model);
2. $Y_{\mathcal{D}} \sim \hat{p} \cdot \hat{\rho}_{\mathcal{E}}$ (approximated model) marked in blue;
3. $Y_{H_0} \sim (q \text{ under } H_0) = \hat{p}$ marked in orange;
4. $Y_{H_1} \sim (q \text{ under } H_1) = \hat{p}'$ marked in green.[6]

We compute LR and $\hat{\text{ASC}}$ statistics for each set and for both density ratios $\{\hat{\rho}, \hat{\rho}_{\mathcal{E}}\}$. The above procedure is repeated for $R = 250$ times and we report empirical distributions of these statistics.

We demonstrate results for MoG-8 with KBC-based DRE and LR statistics. More extensive experiments with k-NN classifiers, ASC statistics, and other hyper-parameters are provided in Appendix D.1. Results for CKB-8 are qualitatively similar to MoG-8 and are provided in Appendix D.2.

We investigate **question 1** (DRE Approximations) in two ways. First, we compare $\hat{\rho}_{\mathcal{E}}$ and $\hat{\rho}$ in Fig. 3. We find that both KBC and k-NN lead to good approximations. We then conduct Kolmogorov–Smirnov (KS) tests between the distributions of $\text{LR}(Y_{H_0}, \hat{\rho})$ vs $\text{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$. If $\hat{\rho} \approx \hat{\rho}_{\mathcal{E}}$ on $\text{supp}(\hat{p})$ then the KS statistics will be close to 0, meaning the two compared distributions are indistinguishable. In Fig. 4a, we plot KS statistics for KBC with different $\sigma_{\mathcal{C}}$. The KS statistics decrease as $\sigma_{\mathcal{C}}$ gets close to $\sigma_{\mathcal{A}}$. We also find larger λ (where fewer samples are deleted) leads to better estimation, as expected.

We investigate **question 2** (Fast Deletion) by asking whether the approximated model $\hat{\rho}_{\mathcal{E}} \cdot \hat{p}$ and the re-trained model $\hat{p}'$ can be distinguished by the ground truth ratio $\hat{\rho}$. We do this by comparing the distributions of $\text{LR}(Y_{H_1}, \hat{\rho})$ vs $\text{LR}(Y_{\mathcal{D}}, \hat{\rho})$; see qualitative comparisons in Fig. 5a and quantitative results in Fig. 4b. We find for a wide range of classifiers, it is hard to distinguish between approximated and re-trained models, especially when λ is larger.

Finally, we answer **question 3** (Hypothesis Test) by comparing the distributions of $\text{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ vs $\text{LR}(Y_{H_1}, \hat{\rho}_{\mathcal{E}})$: see qualitative comparisons in Fig. 5b and quantitative results in Fig. 4c. We find $\hat{\rho}_{\mathcal{E}}$ can distinguish between samples from pre-trained and re-trained models for a wide range of classifiers. In terms of the size of the deletion set, larger λ makes the two models less distinguishable.

5.2 VDM-based DRE for GAN

Experimental setup. The pre-trained model is a DCGAN [36] trained on the full MNIST and Fashion-MNIST. We construct the deletion set X' by randomly selecting samples with certain labels (see details in Appendix E).

Method and results. We train VDM-based DRE based on (5) introduced in Section 3.2. We set T to be the same architecture as the discriminator.

We investigate **question 2** (Fast Deletion) by comparing label distribution of $m = 50$K generated samples from the re-trained and approximated models. We randomly select $10\% - 40\%$ samples with even or odd labels as the deletion set. For each setting we run experiments with five random seeds. Results for deleting 30% samples with even labels are shown in Fig. 6a-6b, and additional results for different deletion sets are provided in Appendix E. We find approximated models generate fewer samples with even labels, and the label distributions are close to re-trained models.

We investigate **question 3** (Hypothesis Test) similarly to Section 5.1. We draw i.i.d. samples $\hat{Y}, Y_{H_0} \sim \hat{p}$, and $Y_{H_1} \sim \hat{p}'$, each of size $m = 1$K. We then compute LR and $\hat{\text{ASC}}$ statistics for each set with density ratio $\hat{\rho}_{\mathcal{E}}$. This procedure is repeated for $R = 100$ times. We compare distributions of $\text{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ vs $\text{LR}(Y_{H_1}, \hat{\rho}_{\mathcal{E}})$ and $\hat{\text{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ vs $\hat{\text{ASC}}_{\phi}(\hat{Y}, Y_{H_1}, \hat{\rho}_{\mathcal{E}})$ in Fig. 7 and Appendix E. In most cases, $\hat{\rho}_{\mathcal{E}}$ can clearly distinguish samples between pre-trained and re-trained models.

To evaluate generation quality, we evaluate the Inception Score (IS) of pre-trained, re-trained, and approximate deletion model in Appendix E.4. The generation qualities are similar across these models.

Time Complexity Analysis. Empirically, our approximated deletion gives a $10.9\times$ speedup over re-training on MNIST, and a $12.0\times$ speedup on Fashion-MNIST.

The training time complexity can be measured by the number of gradient computations. Let the time complexity of computing the gradients of the generator/discriminator be T_G and T_D. Let N_R be the epochs for retraining, $N_D = N_R/5$ be the epochs for training DRE, and N_I be the number of iterations per epoch. Re-training has complexity $C_R = N_R * N_I * (T_D + T_G/5)$ (as the generator is updated every 5 iterations). Training DRE has complexity $C_D = N_D * N_I * T_D$. The fraction of these two complexities, which is the theoretical speedup, is $C_R/C_D = 5 + T_G/T_D$.

6 Related Work

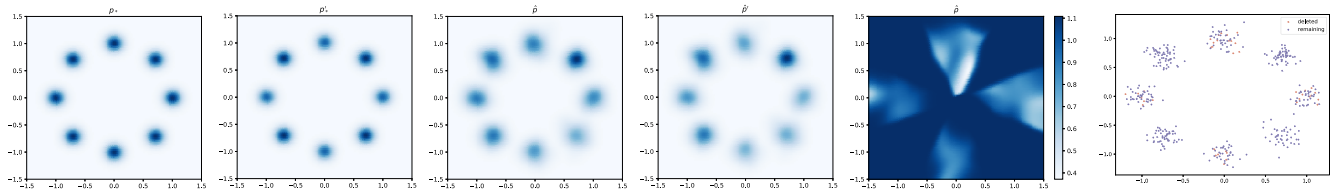
Exact data deletion from learned models (where the altered model is identical to the re-trained model) was introduced as *machine unlearning* [7, 4]. Such deletion can be performed efficiently for relatively simple learners such as linear regression [8] and k-nearest neighbors [39]. Machine unlearning for convex risk minimization was shown theoretically possible under total variation stability [44]. Others have introduced further definitions of approximate data deletion [17, 31, 40, 18] and developed efficient methods for approximate deletion in supervised learning.

The unsupervised setting has received substantially less attention with respect to data deletion. A notable exception is clustering [12, 3]. Our work instead focuses on generative models where the goal is to learn adistribution from data rather than finding clusters. Potential avenues for future work include forging a deeper connection between approximate data deletion for generative models and differential privacy [11] and using recent advances in *certified removal* [17] for generative models.

Outside of the context of data deletion, density ration estimation seeks to estimate the ratio between two densities from samples. For example, the ratio can be estimated via probabilistic classification [43] or variational divergence minimization [34]. There are also many other techniques in the literature [47, 43, 34, 30, 22, 37, 21, 9, 10], all designed for settings with little prior information about the data. In contrast, we consider a setting where we have strong prior information (the only two possibilities are that X' was or was not deleted, rather

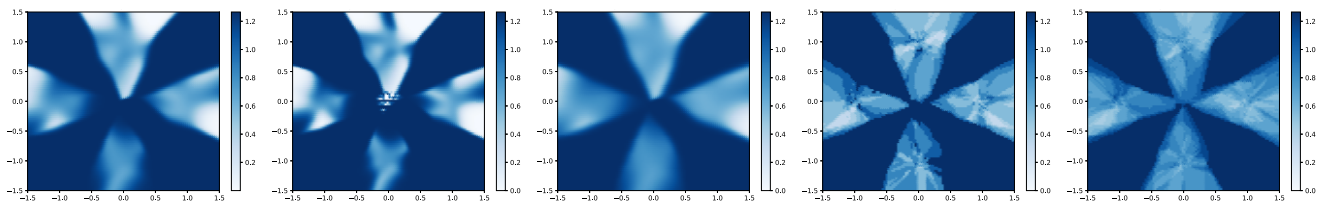
[5] We use non-parametric classifiers because the learning algorithm is non-parametric. In preliminary experiments we found parametric classifiers such as logistic regression are less effective. We conjecture this is due to imbalanced labels, but leave a further investigation as future work.

[6] These colors are used in distribution comparisons and label statistics in Fig. 5, Fig. 6, Fig. 7 and the Appendix.



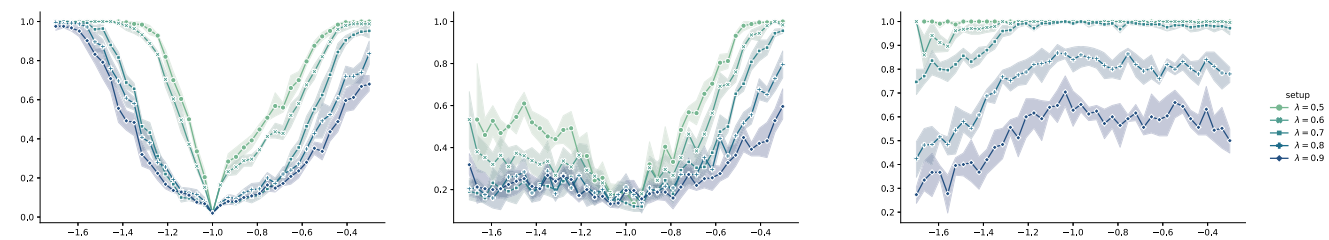
(a) p_* (b) p'_* (c) $\hat{p}$ (d) $\hat{p}'$ (e) $\hat{\rho}$ (f) X' and $X \setminus X'$

Figure 2: Visualization of the experimental setup of MoG-8. (a) Data distribution p_*. (b) Distribution p'_* with $\lambda = 0.8$. (c) Pre-trained KDE $\hat{p}$ on X with $\sigma_{\mathcal{A}} = 0.1$. (d) Re-trained KDE $\hat{p}'$ on $X \setminus X'$ with $\sigma_{\mathcal{A}} = 0.1$. (e) Density ratio $\hat{\rho} = \hat{p}'/\hat{p}$. (f) Deletion set X' and the remaining set $X \setminus X'$.



(a) $\hat{\rho}$ (b) KBC ($\sigma_{\mathcal{C}}$=0.075) (c) KBC ($\sigma_{\mathcal{C}}$=0.125) (d) k-NN (k=10) (e) k-NN (k=20)

Figure 3: Answer to question 1: visualization of ratios in the setting of MoG-8 with $\lambda = 0.6$ and $\sigma_{\mathcal{A}} = 0.1$. (a) $\hat{\rho}$. (b-c) $\hat{\rho}_{\mathcal{E}}$ for KBC-based DRE. (d-e) $\hat{\rho}_{\mathcal{E}}$ for kNN-based DRE. These DREs are visually close to $\hat{\rho}$.



(a) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ (b) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$ (c) $\mathrm{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho}_{\mathcal{E}})$

Figure 4: KS test results (y-axis) between distributions of LR statistics for KBC with different $\log_{10} \sigma_{\mathcal{C}}$ (x-axis). Smaller KS values (y-axis) indicate the two compared distributions are closer. When $\sigma_{\mathcal{C}} \approx \sigma_{\mathcal{A}}$, (a) answers question 1 (DRE Approximations) by showing $\hat{\rho}_{\mathcal{E}} \approx \hat{\rho}$ on the support of $\hat{p}$, (b) answers question 2 (Fast Deletion) by showing Y_{H_1} (from $\hat{p}'$) and $Y_{\mathcal{D}}$ (from the approximated model) cannot be distinguished by $\hat{\rho}$, and (c) answers question 3 (Hypothesis Test) by showing our DRE easily distinguishes Y_{H_0} (from $\hat{p}$) and Y_{H_1} (from $\hat{p}'$). Results for ASC statistics are in Appendix D and are similar to LR.

than in prior work where the two samples can be arbitrarily separated). We adapt probabilistic classification [43] and variational divergence minimization [34] for our setting as they lend themselves naturally to incorporating the knowledge that training data is being deleted. An avenue of future work is incorporating such knowledge into other density ratio estimation methods, any of which can be used within our general framework in Fig. 1.

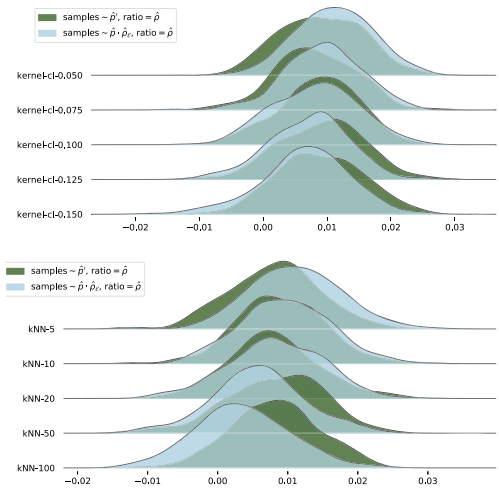
6.1 *Further Discussion of Our Contributions in Relationship to Related Areas*

We discuss our contributions in relationship to three related areas: differential privacy, membership inference, and influence functions.

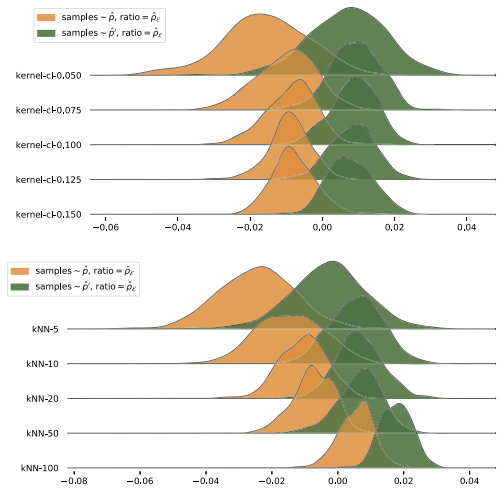
Relationship to differential privacy. First, we note that if the learner is differentially private [11], then the re-trained model is close to the pre-trained model. This means that there is no need to perform data deletion, and it is by definition impossible to test whether training points have been deleted.

Relationship to membership inference. Second, membership inference attackers query whether a particular sample is used for training [42]. This is akin to when the deletion set $X' = \{x'\}$ contains only one sample and membership inference is performed to test whether the training set contains x' or not. In contrast, our deletion test is based on additional prior knowledge and tests whether the training set is X or $X \setminus \{x'\}$. Therefore, membership inference is stronger but harder than the deletion test.

Relationship to influence functions. Finally, we highlight that influence functions [23, 24, 2, 25] designed for likelihood in generative models can potentially be used to estimate density ratio in our framework. The influence function of a sample is a measure of the impact of removing that sample from the training set on the loss function of a particular test sample [23]. When the deletion set only contains one sample, we could use the approximate influence score [25] to derive DRE for deep generative models. We could then generalize to deleting multiple samples by summing individual influences [24, 2], which is another important direction of future work.

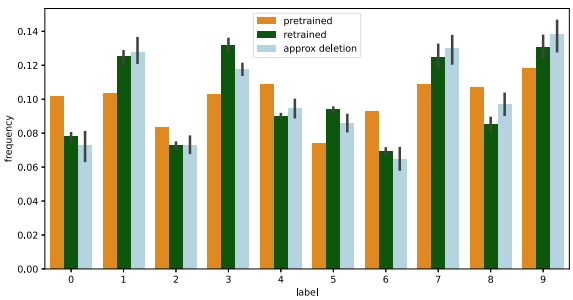


(a) Answer to question 2: these distributions largely overlap with each other, indicating the approximated model cannot be distinguished from the re-trained model.

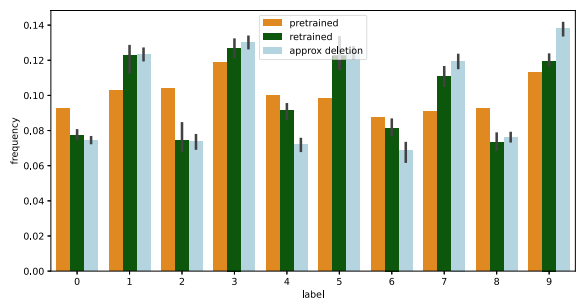


(b) Answer to question 3: these distributions are separated from each other, indicating the DRE can distinguish between samples from pre-trained and re-trained models.

Figure 5: Distributions of (a) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$ and (b) $\mathrm{LR}(Y_{H_0}, \hat{\rho}_\mathcal{E})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho}_\mathcal{E})$. The approximated models and $\hat{\rho}_\mathcal{E}$ are derived from KBC-based DREs with five $\sigma_\mathcal{C}$ values (first row) and kNN-based DREs with five k values (second row). x-axis is LR statistic and y-axis is frequency.



(a) MNIST



(b) Fashion-MNIST

Figure 6: The Fast Deletion question: label distributions of 50K generated samples from pre-trained, re-trained, and approximated models on MNIST (a) Fashion-MNIST (b). Mean and standard errors of five random runs are reported. The label distributions of the approximated model is close to the re-trained model.

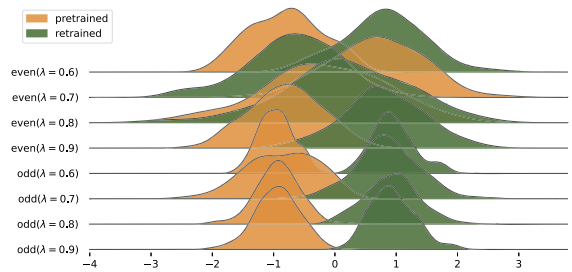


Figure 7: The Hypothesis Test question: distributions of $\mathrm{LR}(Y_{H_0}, \hat{\rho}_\mathcal{E})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho}_\mathcal{E})$ on MNIST. These distributions are separated from each other, indicating the DRE can distinguish between samples from pre-trained and re-trained models.

7 Conclusions and Future Work

In this paper, we propose a density-ratio-based framework for data deletion in generative modeling. Using this framework, we introduce our two main contributions: a fast method for approximate data deletion and a statistical test for estimating whether or not training points have been deleted. We provide formal guarantees for both contributions under various learner assumptions. In addition, we investigate our approximate deletion method and statistical test on real and synthetic datasets for various generative models. Our experiments confirm that (1) our methods accurately approximate the target density ratio, (2) our deletion method efficiently yields a model indistinguishable from the re-trained model, and (3) our hypothesis tests accurately distinguish samples from pre-trained and re-trained models. We highlight a limitation and important future direction: Our density-ratio-based framework results in stability limitations when applied to more complex datasets, as density ratio estimation becomes challenging when data have higher dimensions and more complex patterns.

Acknowledgement

We thank Kamalika Chaudhuri for discussion and helpful feedback. This work was supported by NSF under CNS 1804829 and ARO MURI W911NF2110317.

References

[1] Borja Balle, Giovanni Cherubin, and Jamie Hayes, 'Reconstructing training data with informed adversaries', in *NeurIPS 2021 Workshop Privacy in Machine Learning*, (2021).

[2] Samyadeep Basu, Xuchen You, and Soheil Feizi, 'On second-order group influence functions for black-box predictions', in *International Conference on Machine Learning*, pp. 715–724. PMLR, (2020).

[3] Michele Borassi, Alessandro Epasto, Silvio Lattanzi, Sergei Vassilvitskii, and Morteza Zadimoghaddam, 'Sliding window algorithms for k-clustering problems', *Advances in Neural Information Processing Systems*, **33**, 8716–8727, (2020).

[4] Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot, 'Machine unlearning', in *2021 IEEE Symposium on Security and Privacy (SP)*, pp. 141–159. IEEE, (2021).

[5] Olivier Bousquet and André Elisseeff, 'Stability and generalization', *The Journal of Machine Learning Research*, **2**, 499–526, (2002).

[6] Francesco Paolo Cantelli, *Intorno ad un teorema fondamentale della teoria del rischio*, Tip. degli operai, 1910.

[7] Yinzhi Cao and Junfeng Yang, 'Towards making systems forget with machine unlearning', in *2015 IEEE Symposium on Security and Privacy*, pp. 463–480. IEEE, (2015).

[8] John M Chambers, 'Regression updating', *Journal of the American Statistical Association*, **66**(336), 744–748, (1971).

[9] Kristy Choi, Madeline Liao, and Stefano Ermon, 'Featurized density ratio estimation', in *Uncertainty in Artificial Intelligence*, pp. 172–182. PMLR, (2021).

[10] Kristy Choi, Chenlin Meng, Yang Song, and Stefano Ermon, 'Density ratio estimation via infinitesimal classification', in *International Conference on Artificial Intelligence and Statistics*, pp. 2552–2573. PMLR, (2022).

[11] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith, 'Calibrating noise to sensitivity in private data analysis', in *Theory of cryptography conference*, pp. 265–284. Springer, (2006).

[12] Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou, 'Making ai forget you: Data deletion in machine learning', *Advances in Neural Information Processing Systems*, **32**, (2019).

[13] Ryan Giordano, Michael I Jordan, and Tamara Broderick, 'A higher-order swiss army infinitesimal jackknife', *arXiv preprint arXiv:1907.12116*, (2019).

[14] Ryan Giordano, William Stephenson, Runjing Liu, Michael Jordan, and Tamara Broderick, 'A swiss army infinitesimal jackknife', in *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1139–1147. PMLR, (2019).

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio, 'Generative adversarial nets', *Advances in neural information processing systems*, **27**, (2014).

[16] Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola, 'A kernel two-sample test', *The Journal of Machine Learning Research*, **13**(1), 723–773, (2012).

[17] Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten, 'Certified data removal from machine learning models', *arXiv preprint arXiv:1911.03030*, (2019).

[18] Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou, 'Approximate data deletion from machine learning models', in *International Conference on Artificial Intelligence and Statistics*, pp. 2008–2016. PMLR, (2021).

[19] Takafumi Kanamori, Taiji Suzuki, and Masashi Sugiyama, 'f-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models', *IEEE Transactions on Information Theory*, **58**(2), 708–720, (2011).

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila, 'Analyzing and improving the image quality of stylegan', in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8110–8119, (2020).

[21] Masahiro Kato and Takeshi Teshima, 'Non-negative bregman divergence minimization for deep direct density ratio estimation', in *International Conference on Machine Learning*, pp. 5320–5333. PMLR, (2021).

[22] Haidar Khan, Lara Marcuse, and Bülent Yener, 'Deep density ratio estimation for change point detection', *arXiv preprint arXiv:1905.09876*, (2019).

[23] Pang Wei Koh and Percy Liang, 'Understanding black-box predictions via influence functions', in *International conference on machine learning*, pp. 1885–1894. PMLR, (2017).

[24] Pang Wei W Koh, Kai-Siang Ang, Hubert Teo, and Percy S Liang, 'On the accuracy of influence functions for measuring group effects', *Advances in neural information processing systems*, **32**, (2019).

[25] Zhifeng Kong and Kamalika Chaudhuri, 'Understanding instance-based interpretability of variational auto-encoders', *Advances in Neural Information Processing Systems*, **34**, (2021).

[26] Yann LeCun, Corinna Cortes, and CJ Burges, 'Mnist handwritten digit database', *ATT Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, **2**, (2010).

[27] Friedrich Liese and Igor Vajda, 'On divergences and informations in statistics and information theory', *IEEE Transactions on Information Theory*, **52**(10), 4394–4412, (2006).

[28] Shuang Liu and Kamalika Chaudhuri, 'The inductive bias of restricted f-gans', *arXiv preprint arXiv:1809.04542*, (2018).

[29] Xuejiao Liu, Yao Xu, and Xueshuang Xiang, 'Towards gans' approximation ability', in *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, (2021).

[30] George V Moustakides and Kalliopi Basioti, 'Training neural networks for likelihood/density ratio estimation', *arXiv preprint arXiv:1911.00405*, (2019).

[31] Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi, 'Descent-to-delete: Gradient-based methods for machine unlearning', in *Algorithmic Learning Theory*, pp. 931–962. PMLR, (2021).

[32] Jerzy Neyman and Egon Sharpe Pearson, 'Ix. on the problem of the most efficient tests of statistical hypotheses', *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character*, **231**(694-706), 289–337, (1933).

[33] XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan, 'Estimating divergence functionals and the likelihood ratio by convex risk minimization', *IEEE Transactions on Information Theory*, **56**(11), 5847–5861, (2010).

[34] Sebastian Nowozin, Botond Cseke, and Ryota Tomioka, 'f-gan: Training generative neural samplers using variational divergence minimization', *Advances in neural information processing systems*, **29**, (2016).

[35] Emanuel Parzen, 'On estimation of a probability density function and mode', *The annals of mathematical statistics*, **33**(3), 1065–1076, (1962).

[36] Alec Radford, Luke Metz, and Soumith Chintala, 'Unsupervised representation learning with deep convolutional generative adversarial networks', *arXiv preprint arXiv:1511.06434*, (2015).

[37] Benjamin Rhodes, Kai Xu, and Michael U Gutmann, 'Telescoping density-ratio estimation', *Advances in Neural Information Processing Systems*, **33**, 4905–4916, (2020).

[38] Murray Rosenblatt, 'Remarks on Some Nonparametric Estimates of a Density Function', *The Annals of Mathematical Statistics*, **27**(3), 832 – 837, (1956).

[39] Sebastian Schelter, '" amnesia"-machine learning models that can forget user data very fast.', in *CIDR*, (2020).

[40] Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh, 'Remember what you want to forget: Algorithms for machine unlearning', *Advances in Neural Information Processing Systems*, **34**, (2021).

[41] Robert J Serfling, *Approximation theorems of mathematical statistics*, volume 162, John Wiley & Sons, 2009.

[42] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov, 'Membership inference attacks against machine learning models', in *2017 IEEE symposium on security and privacy (SP)*, pp. 3–18. IEEE, (2017).

[43] Masashi Sugiyama, Taiji Suzuki, and Takafumi Kanamori, *Density ratio estimation in machine learning*, Cambridge University Press, 2012.

[44] Enayat Ullah, Tung Mai, Anup Rao, Ryan A Rossi, and Raman Arora, 'Machine unlearning via algorithmic stability', in *Conference on Learning Theory*, pp. 4126–4142. PMLR, (2021).

[45] Dominik Wied and Rafael Weißbach, 'Consistency of the kernel density estimator: a survey', *Statistical Papers*, **53**(1), 1–21, (2012).

[46] Han Xiao, Kashif Rasul, and Roland Vollgraf, 'Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms', *arXiv preprint arXiv:1708.07747*, (2017).

[47] Makoto Yamada, Taiji Suzuki, Takafumi Kanamori, Hirotaka Hachiya, and Masashi Sugiyama, 'Relative density-ratio estimation for robust distribution comparison', *Advances in neural information processing systems*, **24**, (2011).

A Notation Table

Table 3: Notations used in this paper.

p_*	data distribution
X	training set: N i.i.d. samples from p_*
X'	deletion set: N' samples from X
$\mathcal{A}$	algorithm of the generative model
$\hat{p}$	pretrained generative model on X
$\hat{p}'$	retrained generative model on $X \setminus X'$
p'_*	distribution s.t. $X \setminus X'$ are i.i.d. samples from p'_*
ρ_*	density ratio p'_*/p_*
$\hat{\rho}$	density ratio $\hat{p}'/\hat{p}$
DRE	density ratio estimator
$\hat{\rho}_{\mathcal{E}}$	abbreviation for $\mathrm{DRE}(X, X \setminus X')$; DRE between X and $X \setminus X'$
$\mathcal{D}(\hat{p}, X, X')$	approximate deletion $\hat{\rho}_{\mathcal{E}} \cdot \hat{p}$
q	the distribution to be tested
m	number of samples drawn from models
Y	m i.i.d. samples from q
Y_{H_i}	m i.i.d. samples from q under H_i, $i = 1, 2$
$\hat{Y}$	m i.i.d. samples from the pretrained model $\hat{p}$
$Y_{\mathcal{D}}$	m i.i.d. samples from the approximate deletion $\mathcal{D}(\hat{p}, X, X')$
R	number of repeats for each statistic
MMD^2	squared MMD metric
$\hat{\mathrm{MMD}}^2_u$	unbiased MMD estimator
LR	likelihood ratio
D_ϕ	the ϕ-divergence
$\hat{\mathrm{ASC}}_\phi$	ASC statistic, or the ϕ-divergence estimator
IF	influence functions
$\tilde{\mathrm{IF}}$	influence function estimator
C	number of checkpoints to compute $\tilde{\mathrm{IF}}$
η	learning rate to compute $\tilde{\mathrm{IF}}$
λ	parameter used to define ρ_* in 2d experiments
KDE	kernel density estimator
KBC	kernel-based classifier
kNN	k nearest neighbour classifier
σ	bandwidth used to define kernel $\mathcal{N}(0, \sigma^2 I)$ in KDE
$\sigma_{\mathcal{A}}$	bandwidth of the learning algorithm in 2d experiments
$\sigma_{\mathcal{C}}$	bandwidth of KBC in 2d experiments

B Theory for the Framework in Section 2

B.1 Omitted Proofs in Section 2.2

Proof of Thm. 2

Proof. Notice that

$$\hat{\rho} = \frac{\hat{p}'}{\hat{p}} = \frac{\hat{p}'}{p'_*} \cdot \frac{p'_*}{p_*} \cdot \frac{p_*}{\hat{p}}.$$

With probability at least $1 - \delta_N$

$$\frac{1}{c_N} \leq \frac{p_*}{\hat{p}} \leq c_N.$$

With probability at least $1 - \delta_{N-N'}$

$$\frac{1}{c_{N-N'}} \leq \frac{\hat{p}'}{p'_*} \leq c_{N-N'}.$$

Therefore, with probability at least $1 - \delta_N - \delta_{N-N'}$,

$$\begin{aligned}\int_{\mathbb{R}^d} \hat{p} \left|\hat{\rho} - \rho_*\right| dx &= \int_{\mathbb{R}^d} p'_* \left|\frac{\hat{p}'}{p'_*} - \frac{\hat{p}}{p_*}\right| dx \\ &\leq \max\left(c_N - \frac{1}{c_{N-N'}}, c_{N-N'} - \frac{1}{c_N}\right) \\ &\leq 2(c_N + c_{N-N'} - 2).\end{aligned}$$

Now, we choose a fixed RC algorithm $\mathcal{A}_0$, and define $\hat{\rho}_{\mathcal{E}}(Z_1, Z_2) = \rho(p_{\mathcal{A}_0(Z_1)}, p_{\mathcal{A}_0(Z_2)})$. Then, with probability at least $1 - \delta_N - \delta_{N-N'}$,

$$\int_{\mathbb{R}^d} \hat{p} \left|\hat{\rho}_{\mathcal{E}} - \rho_*\right| dx \leq 2(c_N + c_{N-N'} - 2).$$

Therefore, with probability at least $1 - 2\delta_N - 2\delta_{N-N'}$,

$$\|\hat{\rho}_{\mathcal{E}} \cdot \hat{p} - \hat{p}'\|_1 = \int_{\mathbb{R}^d} \hat{p} \left|\hat{\rho}_{\mathcal{E}} - \hat{\rho}\right| dx \leq 4(c_N + c_{N-N'} - 2).$$

□

Proof of Thm. 4

Proof. Notice that

$$\begin{aligned}\int_{\mathbb{R}^d} \hat{p}^2 \left|\hat{\rho} - \rho_*\right| dx &= \int_{\mathbb{R}^d} \hat{p} \left|\hat{p}' - \rho_* \hat{p}\right| dx \\ &= \int_{\mathbb{R}^d} \hat{p} \left|\hat{p}' - p'_* + p'_* - \rho_*(\hat{p} - p_* + p_*)\right| dx \\ &= \int_{\mathbb{R}^d} \hat{p} \left|\hat{p}' - p'_* - \rho_*(\hat{p} - p_*)\right| dx.\end{aligned}$$

With probability at least $1 - \delta_N$, $|\hat{p} - p_*| \leq \epsilon_N$, and with probability at least $1 - \delta_{N-N'}$, $|\hat{p}' - p'_*| \leq \epsilon_{N-N'}$. Therefore, with probability at least $1 - \delta_N - \delta_{N-N'}$,

$$\int_{\mathbb{R}^d} \hat{p}^2 \left|\hat{\rho} - \rho_*\right| dx \leq \epsilon_{N-N'} + \|\rho_*\|_\infty \epsilon_N.$$

Now, we choose a fixed TVC algorithm $\mathcal{A}_0$, and define $\hat{\rho}_{\mathcal{E}}(Z_1, Z_2) = \rho(p_{\mathcal{A}_0(Z_1)}, p_{\mathcal{A}_0(Z_2)})$. Then, with probability at least $1 - \delta_N - \delta_{N-N'}$,

$$\int_{\mathbb{R}^d} \hat{p}^2 \left|\hat{\rho}_{\mathcal{E}} - \rho_*\right| dx \leq \epsilon_{N-N'} + \|\rho_*\|_\infty \epsilon_N.$$

Therefore, with probability at least $1 - 2\delta_N - 2\delta_{N-N'}$,

$$\|\hat{\rho}_{\mathcal{E}} \cdot \hat{p} - \hat{p}'\|_{1,\hat{p}} = \int_{\mathbb{R}^d} \hat{p}^2 \left|\hat{\rho} - \hat{\rho}_{\mathcal{E}}\right| dx \leq 2\left(\epsilon_{N-N'} + \|\rho_*\|_\infty \epsilon_N\right).$$

□

B.2 Omitted Proofs in Section 2.3

Proof of Thm. 8

Proof. By taking $Z_0 = Z$, $Z_1 = Z \setminus \{z\}$, and $\hat{Z} = \{\hat{z}\}$, we conclude ϵ-DP implies ϵ-US. By taking one side of the ϵ-US bound, we conclude ϵ-US implies ϵ-LBLI.

Define

$$\hat{\rho}_k = \frac{p_{\mathcal{A}(X \setminus X'_{1:k-1})}}{p_{\mathcal{A}(X \setminus X'_{1:k})}}$$

for $k = 1, \cdots, N'$. Then, ϵ-LBLI indicates $\log \|\hat{\rho}_k\|_\infty \leq \epsilon$. Notice that

$$\hat{\rho} = \prod_{k=1}^{N'} \hat{\rho}_k.$$

Therefore, we have $\log \|\hat{\rho}\|_\infty \leq N'\epsilon$. □

Proof of Thm. 10

Proof. With probability at least $1 - \delta_N$,

$$-\log c_N \leq \log \rho(\mu_i, p_{\mathcal{A}(Z_i)}) \leq \log c_N.$$

Therefore, with probability at least $1 - 2\delta_N$,

$$\left\|\log \rho(\mu_1, p_{\mathcal{A}(Z_1)}) - \log \rho(\mu_2, p_{\mathcal{A}(Z_2)})\right\|_\infty \leq 2 \log c_N.$$

□

Proof of Thm. 11

Proof. Define $Z_k = X \setminus X'_{1:k}$ and μ_k be the distribution such that Z_k contains i.i.d. samples from μ_k. Specifically, $\mu_0 = p_*$ and $\mu_{N'} = p'_*$. Then, we have

$$\begin{aligned}
\log \rho_* - \log \hat{\rho} &= \log \frac{\mu_{N'}}{\mu_0} - \log \frac{p_{\mathcal{A}(Z_{N'})}}{p_{\mathcal{A}(Z_0)}} \\
&= \sum_{k=1}^{N'} \left(\log \frac{\mu_k}{\mu_{k-1}} - \log \frac{p_{\mathcal{A}(Z_k)}}{p_{\mathcal{A}(Z_{k-1})}} \right) \\
&= \sum_{k=1}^{N'} \left(\log \rho(\mu_{k-1}, p_{\mathcal{A}(Z_{k-1})}) - \log \rho(\mu_k, p_{\mathcal{A}(Z_k)}) \right).
\end{aligned}$$

Therefore, with probability at least $1 - N'\delta$, we have

$$\|\log \rho_* - \log \hat{\rho}\|_\infty \leq N'\epsilon,$$

which indicates $\log \|\hat{\rho}\|_\infty \leq N'\epsilon + \log \|\rho_*\|_\infty$. □

Proof of Thm. 13

Proof. By rewriting ES for $\hat{p}$ and $\hat{p}'$, we have

$$\mathbb{E}_{x \sim \hat{p}} \log \hat{\rho} = -\mathbb{KL}\left(\hat{p} \| \hat{p}'\right),$$

$$\mathbb{E}_{x \sim \hat{p}} (\log \hat{\rho})^2 \leq \epsilon.$$

Because

$$\mathbb{E}_{x \sim \hat{p}} (\log \hat{\rho})^2 \geq \left(\mathbb{E}_{x \sim \hat{p}} \log \hat{\rho}\right)^2,$$

we have $\mathbb{KL}\left(\hat{p} \| \hat{p}'\right) \leq \sqrt{\epsilon}$. Then, according to Cantelli's inequality [6], for any positive a,

$$\text{Prob}\left(\log \hat{\rho} \geq -\mathbb{KL}\left(\hat{p} \| \hat{p}'\right) + a\right) \leq \frac{\mathbb{VAR}(\log \hat{\rho})}{\mathbb{VAR}(\log \hat{\rho}) + a^2}.$$

By letting

$$a = \sqrt{\frac{1-\delta}{\delta} \cdot \mathbb{VAR}(\log \hat{\rho})},$$

we have with probability at least $1-\delta$ for samples $x \sim \hat{p}$,

$$\begin{aligned}\log \hat{\rho}(x) &\leq \sqrt{\frac{1-\delta}{\delta} \cdot \left(\mathbb{E}_{x\sim\hat{p}}(\log \hat{\rho})^2 - \mathbb{KL}\left(\hat{p}\|\hat{p}'\right)^2\right)} - \mathbb{KL}\left(\hat{p}\|\hat{p}'\right) \\ &\leq \sqrt{\frac{\epsilon(1-\delta)}{\delta}}.\end{aligned}$$

□

C Statistical Tests in Section 4

C.1 Likelihood Ratio Tests

Proof of Thm. 18

Proof. By definition of RC, we have with probability at least $1-\delta_N$,

$$|\log\hat{p}-\log p_*|\leq\log c_N,$$

and with probability at least $1-\delta_{N-N'}$,

$$|\log\hat{p}'-\log p'_*|\leq\log c_{N-N'}.$$

Therefore, with probability at least $1-\delta_N-\delta_{N-N'}$,

$$|\log\hat{\rho}-\log\rho_*|\leq\log c_{N-N'}+\log c_N.$$

Now, we choose a fixed RC algorithm $\mathcal{A}_0$, and define $\hat{\rho}_{\mathcal{E}}(Z_1,Z_2)=\rho(p_{\mathcal{A}_0(Z_1)},p_{\mathcal{A}_0(Z_2)})$. Then, we also have with probability at least $1-\delta_N-\delta_{N-N'}$,

$$|\log\hat{\rho}_{\mathcal{E}}-\log\rho_*|\leq\log c_{N-N'}+\log c_N.$$

Therefore, with probability at least $1-2(\delta_N+\delta_{N-N'})$,

$$|\log\hat{\rho}-\log\hat{\rho}_{\mathcal{E}}|\leq 2(\log c_{N-N'}+\log c_N),$$

and the conclusion follows. □

Proof of Thm. 19

Proof. (1) Notice that

$$\begin{aligned}|\mathrm{LR}(Y,\hat{p},\hat{p}')-\mathrm{LR}(Y,\hat{p},\hat{\rho}_{\mathcal{E}}\cdot\hat{p})| &= \frac{1}{m}\sum_{y\in Y}|\log\hat{\rho}(y)-\log\hat{\rho}_{\mathcal{E}}(y)|\\ &\leq\frac{1}{m}\cdot m\epsilon\\ &=\epsilon.\end{aligned}$$

(2) If H_0 is true, then $Y\sim\hat{p}$. Then,

$$\begin{aligned}\mathbb{E}_Y|\mathrm{LR}(Y,\hat{p},\hat{p}')-\mathrm{LR}(Y,\hat{p},\hat{\rho}_{\mathcal{E}}\cdot\hat{p})| &= \mathbb{E}_Y\left|\frac{1}{m}\sum_{y\in Y}(\log\hat{\rho}(y)-\log\hat{\rho}_{\mathcal{E}}(y))\right|\\ &\leq\mathbb{E}_Y\left(\frac{1}{m}\sum_{y\in Y}|\log\hat{\rho}(y)-\log\hat{\rho}_{\mathcal{E}}(y)|\right)\\ &=\mathbb{E}_{y\sim\hat{p}}|\log\hat{\rho}(y)-\log\hat{\rho}_{\mathcal{E}}(y)|\\ &=\|\log\hat{\rho}-\log\hat{\rho}_{\mathcal{E}}\|_{1,\hat{p}}\\ &\leq\epsilon.\end{aligned}$$

By Markov's inequality, we have with probability at least $1-\delta$, $|\mathrm{LR}(Y,\hat{p},\hat{p}')-\mathrm{LR}(Y,\hat{p},\hat{\rho}_{\mathcal{E}}\cdot\hat{p})|\leq\epsilon/\delta$. The proof for H_1 is similar. □

Statistical properties of LR **statistics.** Let $\phi(t)=\log(t)^2$. When H_0 is true, we have

$$\begin{aligned}\mathbb{E}_{Y\sim\hat{p}}\ \mathrm{LR}(Y,\hat{p},\hat{p}') &= \mathbb{E}_{\hat{p}}\log\frac{\hat{p}'}{\hat{p}}=-\mathbb{KL}\left(\hat{p}\|\hat{p}'\right),\\ \mathbb{VAR}_{Y\sim\hat{p}}\ \mathrm{LR}(Y,\hat{p},\hat{p}') &= \frac{1}{m}\left(\mathbb{E}_{\hat{p}}\left(\log\frac{\hat{p}'}{\hat{p}}\right)^2-\left(\mathbb{E}_{\hat{p}}\log\frac{\hat{p}'}{\hat{p}}\right)^2\right)\\ &=\frac{1}{m}\left(D_{\log^2}(\hat{p}\|\hat{p}')-\mathbb{KL}\left(\hat{p}\|\hat{p}'\right)^2\right).\end{aligned}$$

When H_1 is true, we have

$$\begin{aligned}\mathbb{E}_{Y\sim\hat{p}'}\ \mathrm{LR}(Y,\hat{p},\hat{p}') &= \mathbb{E}_{\hat{p}'}\log\frac{\hat{p}'}{\hat{p}}=\mathbb{KL}\left(\hat{p}'\|\hat{p}\right),\\ \mathbb{VAR}_{Y\sim\hat{p}'}\ \mathrm{LR}(Y,\hat{p},\hat{p}') &= \frac{1}{m}\left(\mathbb{E}_{\hat{p}'}\left(\log\frac{\hat{p}'}{\hat{p}}\right)^2-\left(\mathbb{E}_{\hat{p}'}\log\frac{\hat{p}'}{\hat{p}}\right)^2\right)\\ &=\frac{1}{m}\left(D_{\log^2}(\hat{p}'\|\hat{p})-\mathbb{KL}\left(\hat{p}'\|\hat{p}\right)^2\right).\end{aligned}$$

C.2 ASC Tests

Proof of Thm. 20

Proof. Take expectations $Y \sim q$ and $\hat{Y} \sim \hat{p}$. Then, we have

$$\begin{aligned}
\mathbb{E}|\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) - \hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}_\mathcal{E})| &= \mathbb{E}\left|\frac{1}{m}\left(\sum_{y\in\hat{Y}} + \sum_{y\in Y}\right)(\psi(\hat{\rho}(y)) - \psi(\hat{\rho}_\mathcal{E}(y)))\right| \\
&\leq \mathbb{E}\left(\frac{1}{m}\sum_{y\in\hat{Y}}|\psi(\hat{\rho}(y)) - \psi(\hat{\rho}_\mathcal{E}(y))|\right) \\
&+ \mathbb{E}\left(\frac{1}{m}\sum_{y\in Y}|\psi(\hat{\rho}(y)) - \psi(\hat{\rho}_\mathcal{E}(y))|\right) \\
&= \mathbb{E}_{y\sim\hat{p}}|\psi(\hat{\rho}(y)) - \psi(\hat{\rho}_\mathcal{E}(y))| + \mathbb{E}_{y\sim q}|\psi(\hat{\rho}(y)) - \psi(\hat{\rho}_\mathcal{E}(y))| \\
&= \|\psi(\hat{\rho}) - \psi(\hat{\rho}_\mathcal{E})\|_{1,\hat{p}} + \|\psi(\hat{\rho}) - \psi(\hat{\rho}_\mathcal{E})\|_{1,q} \\
&\leq 2\epsilon.
\end{aligned}$$

By Markov's inequality, we have with probability at least $1-\delta$, it holds that $|\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) - \hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}_\mathcal{E})| \leq 2\epsilon/\delta$.

□

Statistical properties of ASC **statistics.** When H_0 is true, we have

$$\mathbb{E}_{Y\sim\hat{p},\hat{Y}\sim\hat{p}}\ \hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) = \mathbb{E}_{\hat{p}}\left(\frac{2\phi(\hat{\rho}(y))}{1+\hat{\rho}(y)}\right).$$

When H_1 is true, we have

$$\begin{aligned}
\mathbb{E}_{Y\sim\hat{p}',\hat{Y}\sim\hat{p}}\ \hat{\mathrm{ASC}}_\phi(\hat{Y}, Y, \hat{\rho}) &= (\mathbb{E}_{\hat{p}} + \mathbb{E}_{\hat{p}'})\frac{\phi(\hat{\rho})}{1+\hat{\rho}} \\
&= \mathbb{E}_{\hat{p}}(1+\hat{\rho})\cdot\frac{\phi(\hat{\rho})}{1+\hat{\rho}} \\
&= \mathbb{E}_{\hat{p}}\left(\phi(\hat{\rho}(y))\right).
\end{aligned}$$

C.3 MMD Tests

Definition of MMD. Let $K_{\mathrm{MMD}}(\cdot,\cdot)$ be a kernel function. The Maximum Mean Discrepancy (MMD) [16] between $\hat{p}$ and q is defined as

$$\mathrm{MMD}^2(q,\hat{p}) = \left(\mathbb{E}_{x,y\sim\hat{p}} - 2\mathbb{E}_{x\sim\hat{p},y\sim q} + \mathbb{E}_{x,y\sim q}\right)K_{\mathrm{MMD}}(x,y).$$

Given m i.i.d. samples $\hat{Y}\sim\hat{p}$ and m i.i.d. samples $Y\sim q$, an unbiased estimator of MMD^2 is

$$\hat{\mathrm{MMD}}^2_u(Y,\hat{Y}) = \frac{1}{m(m-1)}\sum_{i\neq j}(K_{\mathrm{MMD}}(y_i,y_j) + K_{\mathrm{MMD}}(\hat{y}_i,\hat{y}_j)) - \frac{2}{m^2}\sum_{i,j}K_{\mathrm{MMD}}(y_i,\hat{y}_j).$$

Asymptotic and concentration properties[41, 16]. Define

$$h((y_i,\hat{y}_i),(y_j,\hat{y}_j)) = K_{\mathrm{MMD}}(y_i,y_j) + K_{\mathrm{MMD}}(\hat{y}_i,\hat{y}_j) - K_{\mathrm{MMD}}(y_i,\hat{y}_j) - K_{\mathrm{MMD}}(y_j,\hat{y}_i).$$

Then, we have

$$\hat{\mathrm{MMD}}^2_u(Y,\hat{Y}) = \frac{1}{m(m-1)}\sum_{i\neq j}^{m}h((y_i,\hat{y}_i),(y_j,\hat{y}_j)).$$

Define

$$\begin{aligned}
\sigma_u^2 &= 4\left(\mathop{\mathbb{E}_{y\sim q}}_{\hat{y}\sim\hat{p}}\left[\mathop{\mathbb{E}_{y'\sim q}}_{\hat{y}'\sim\hat{p}}h((y,\hat{y}),(y',\hat{y}'))\right]^2 - \left[\mathop{\mathbb{E}_{y,y'\sim q}}_{\hat{y},\hat{y}'\sim\hat{p}}h((y,\hat{y}),(y',\hat{y}'))\right]^2\right) \\
&= 4\cdot\mathop{\mathbb{E}_{y\sim q}}_{\hat{y}\sim\hat{p}}\mathbb{VAR}_{\substack{y'\sim q\\ \hat{y}'\sim\hat{p}}}\ h((y,\hat{y}),(y',\hat{y}')).
\end{aligned}$$

Then, it holds that

$$\sqrt{m}\left(\hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) - \mathrm{MMD}^2(q,\hat{p})\right) \to \mathcal{N}(0,\sigma_u^2) \text{ in distribution}$$

As for concentration properties, with probability at least $1-\delta$, it holds that

$$\mathrm{MMD}_u^2(Y,\hat{Y}) - \mathrm{MMD}^2(q,\hat{p}) \le 4\sqrt{\frac{1}{m}\log\frac{1}{\delta}} \cdot \sup_{x,y} K_{\mathrm{MMD}}(x,y),$$

with have the same bound on the other side.

Asymptotic and concentration properties in the context of deletion test. Now, we look at these properties in the context of deletion test. If H_0 is true,

$$\begin{aligned}
\mathbb{E}_{Y\sim\hat{p}}\ \hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) &= 0,\\
\mathbb{VAR}_{Y\sim\hat{p}}\ \hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) &= \frac{4}{m}\cdot \mathbb{E}_{\substack{y\sim\hat{p}\\ \hat{y}\sim\hat{p}}} \mathbb{VAR}_{\substack{y'\sim\hat{p}\\ \hat{y}'\sim\hat{p}}}\ h((y,\hat{y}),(y',\hat{y}')).
\end{aligned}$$

And with probability at least $1-\delta$,

$$\left|\hat{\mathrm{MMD}}_u^2(Y,\hat{Y})\right| \le 4\sqrt{\frac{1}{m}\log\frac{2}{\delta}} \cdot \sup_{x,y} K_{\mathrm{MMD}}(x,y).$$

If H_1 is true,

$$\begin{aligned}
\mathbb{E}_{Y\sim\hat{p}'}\ \hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) &= \mathrm{MMD}^2(\hat{p}',\hat{p}),\\
\mathbb{VAR}_{Y\sim\hat{p}}\ \hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) &= \frac{4}{m}\cdot \mathbb{E}_{\substack{y\sim q\\ \hat{y}\sim\hat{p}}} \mathbb{VAR}_{\substack{y'\sim q\\ \hat{y}'\sim\hat{p}}}\ h((y,\hat{y}),(y',\hat{y}')).
\end{aligned}$$

And with probability at least $1-\delta$,

$$\left|\hat{\mathrm{MMD}}_u^2(Y,\hat{Y}) - \mathrm{MMD}^2(\hat{p}',\hat{p})\right| \le 4\sqrt{\frac{1}{m}\log\frac{2}{\delta}} \cdot \sup_{x,y} K_{\mathrm{MMD}}(x,y).$$

Example 21 (KDE)**.** *Now, we compute* $\mathrm{MMD}(\hat{p}',\hat{p})^2$ *for KDE with the standard Gaussian kernel. We let* K_{MMD} *be the standard RBF kernel:* $K_{\mathrm{MMD}}(x,y) = \exp(-\|x-y\|^2/2)$*. Let* $x,x' \sim q$*,* $y,y' \sim \hat{p}$*, and* $z_i, z_i' \sim \mathcal{N}(x_i, I)$*. Then,*

$$\begin{aligned}
\mathbb{E}_{x,x'} K_{\mathrm{MMD}}(x,x') &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} \mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j')\\
\mathbb{E}_{y,y'} K_{\mathrm{MMD}}(y,y') &= \frac{1}{(N-N')^2}\sum_{i=N'+1}^{N}\sum_{j=N'+1}^{N} \mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j')\\
\mathbb{E}_{x,y} K_{\mathrm{MMD}}(x,y) &= \frac{1}{N(N-N')}\sum_{i=1}^{N}\sum_{j=N'+1}^{N} \mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j').
\end{aligned}$$

By rearranging, we have

$$\mathrm{MMD}^2(\hat{p}',\hat{p}) = \left(\frac{N'^2}{N^2(N-N')^2}\sum_{i=N'+1}^{N}\sum_{j=N'+1}^{N} - \frac{N+N'}{N^2(N-N')}\sum_{i=1}^{N'}\sum_{j=N'+1}^{N} + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N'}\right) \mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j').$$

We then compute $\mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j')$*.*

$$\begin{aligned}
\mathbb{E}_{z_i,z_j'} K_{\mathrm{MMD}}(z_i,z_j') &= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} \mathcal{N}(z_i;x_i,I)\mathcal{N}(z_j';x_j,I) K_{\mathrm{MMD}}(z_i,z_j') dz_i dz_j'\\
&= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} \frac{1}{(2\pi)^d}\exp\left(-\frac{\|z_i-x_i\|^2 + \|z_j'-x_j\|^2}{2} - \frac{\|z_i-z_j'\|^2}{2}\right) dz_i dz_j'.
\end{aligned}$$

We apply a change-of-variable formula:

$$\begin{aligned}
z_i &= -\frac{v_i}{\sqrt{2}} - \frac{v_j'}{\sqrt{6}} + \frac{2}{3}x_i + \frac{1}{3}x_j,\\
z_j &= -\frac{v_i}{\sqrt{2}} + \frac{v_j'}{\sqrt{6}} + \frac{1}{3}x_i + \frac{2}{3}x_j.
\end{aligned}$$

Then,

$$\frac{\|z_i - x_i\|^2 + \|z'_j - x_j\|^2}{2} + \frac{\|z_i - z'_j\|^2}{2} = \frac{1}{2}\left(\|v_i\|^2 + \|v'_j\|^2 + \frac{\|x_i - x_j\|^2}{3}\right).$$

Therefore,

$$\begin{aligned}\mathbb{E}_{z_i, z'_j} K_{\mathrm{MMD}}(z_i, z'_j) &= \int_{\mathbb{R}^d}\int_{\mathbb{R}^d} \frac{1}{(2\pi)^d} \exp\left(-\frac{\|v_i\|^2 + \|v'_j\|^2}{2} - \frac{\|x_i - x_j\|^2}{6}\right) \left|\det\left(\frac{\partial(z_i, z'_j)}{\partial(v_i, v'_j)}\right)\right|^d dv_i dv'_j \\ &= 3^{-\frac{d}{2}} \exp\left(-\frac{\|x_i - x_j\|^2}{6}\right).\end{aligned}$$

Summing up, we have

$$\mathrm{MMD}^2(\hat{p}', \hat{p}) = 3^{-\frac{d}{2}}\left(\frac{N'^2}{N^2(N-N')^2}\sum_{i=N'+1}^{N}\sum_{j=N'+1}^{N} - \frac{N+N'}{N^2(N-N')}\sum_{i=1}^{N'}\sum_{j=N'+1}^{N} + \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N'}\right)\exp\left(-\frac{\|x_i - x_j\|^2}{6}\right).$$

D Experiments on Two-Dimensional Synthetic Datasets

D.1 MoG-8

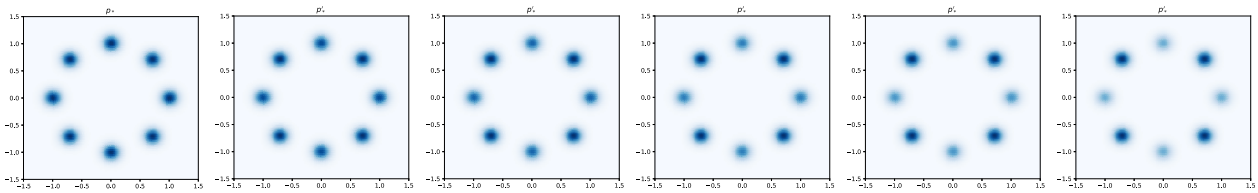
Setup. The MoG-8 data distribution is defined as

$$p_*(x) = \frac{1}{8}\sum_{i=1}^{8}\mathcal{N}(x; (\cos\theta_i, \sin\theta_i), 0.1I),$$

where $\theta_i = \frac{2\pi i}{8}$. The modified distribution p'_* with weight λ is defined as

$$p'_*(x) = \frac{1}{4(1+\lambda)}\sum_{i=1}^{8} w_i\mathcal{N}(x; (\cos\theta_i, \sin\theta_i), 0.1I),$$

where $w_i = 1$ for even i and λ for odd i. The construction algorithm for X is randomly sampling a cluster id between 1 and 8 and randomly drawing a sample from the corresponding Gaussian distribution. The construction algorithm for X' is to include a sample $x \in X$ with probability $1 - \lambda$ if x is from i-th Gaussian for odd i. The distributions and data with different λ are shown in Fig. 8.



(a) p_* (b) $p'_*(\lambda = 0.9)$ (c) $p'_*(\lambda = 0.8)$ (d) $p'_*(\lambda = 0.7)$ (e) $p'_*(\lambda = 0.6)$ (f) $p'_*(\lambda = 0.5)$

Figure 8: Visualization of the experimental setup of MoG-8. (a) Data distribution p_*. (b) - (f) p'_* with different λ values. A larger λ means less data is deleted.

Other hyperparameters are set as follows. The number of training samples $N = 400$ unless specified. The number of samples for the deletion test $m = 400$ unless specified. The number of repeats for each setup is $R = 250$ unless specified. The learning algorithm KDE has bandwidth $\sigma_{\mathcal{A}} = 0.1$ unless specified.

Question 1 (DRE Approximations). We visualize $\hat{\rho}$ and $\hat{\rho}_{\mathcal{E}}$ in Fig. 9 (extension of Fig. 3). These figures give qualitative answers to question 1.

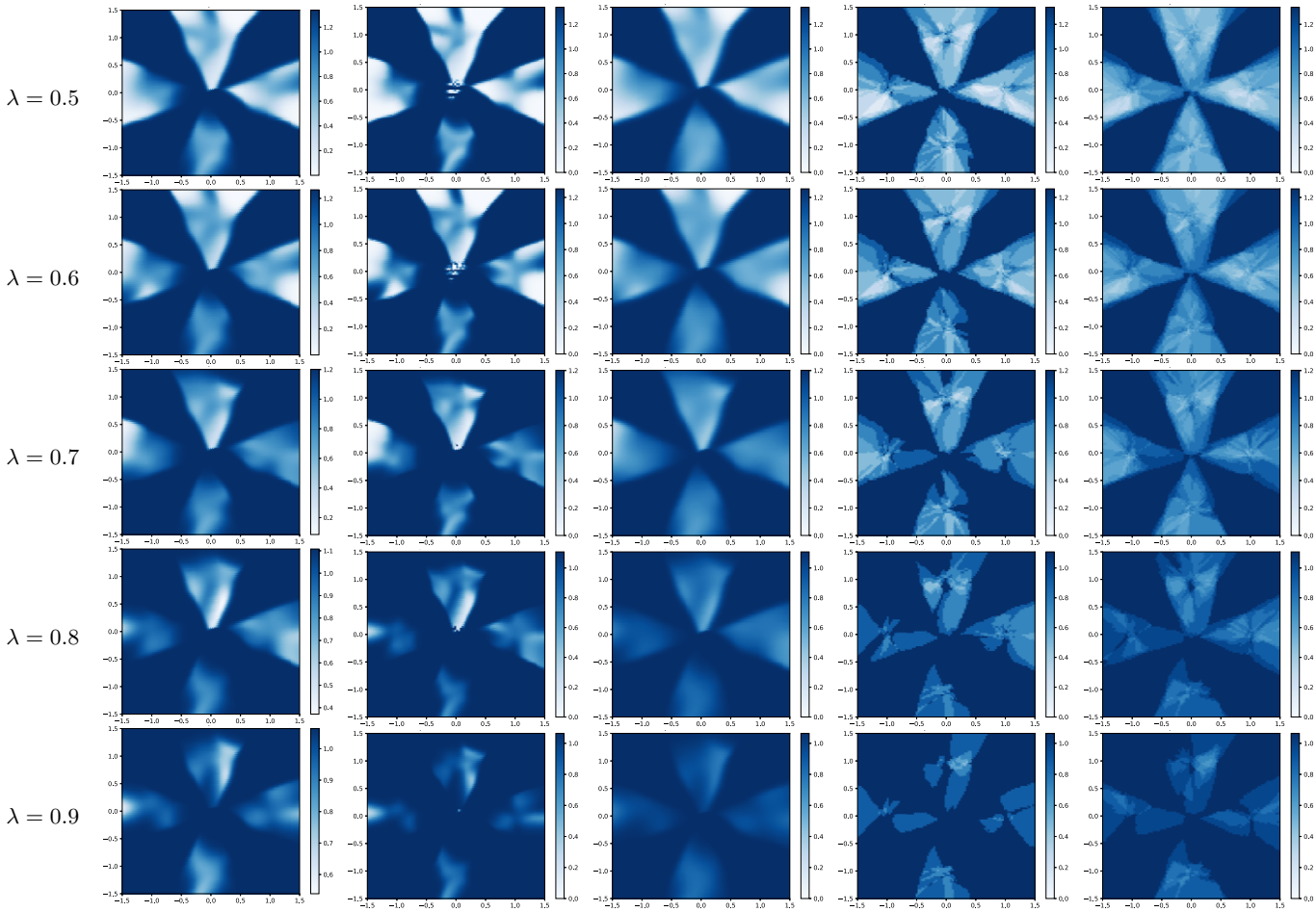
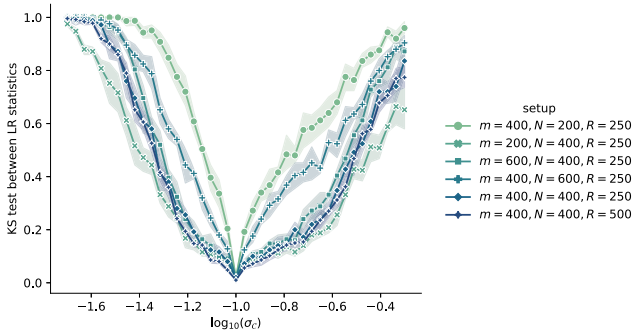
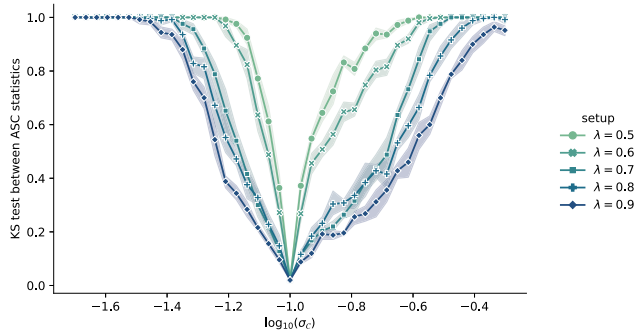


Figure 9: Visualization of ratio $\hat{\rho}$ in (a) and $\hat{\rho}_{\mathcal{E}}$ in (b)-(e) for different classifier-based DREs.

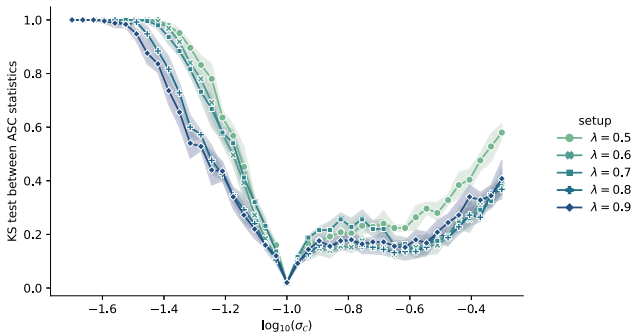
We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 10 (extension of Fig. 4a). When $\sigma_{\mathcal{C}} \approx \sigma_{\mathcal{A}} = 0.1$, the KS values are small, indicating KBC with these $\sigma_{\mathcal{C}}$ can lead to classifier-based DRE $\hat{\rho}_{\mathcal{E}}$ that is close to $\hat{\rho}$. In terms of statistics, the estimation is most accurate under KL and least accurate under Hellinger distance. In terms of λ, the estimation is more accurate for larger λ, where less data are deleted, as expected.



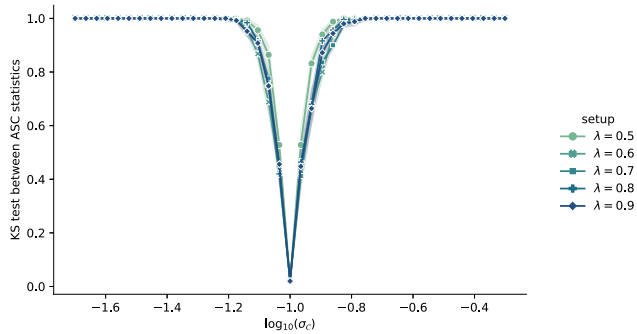
(a) LR statistics with $\lambda = 0.8$ and different m, N, R



(b) ASC statistics with $\phi(t) = \log(t)$



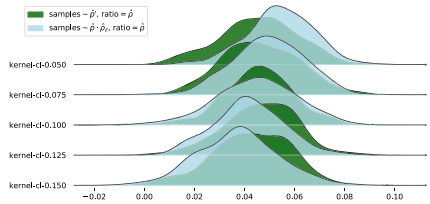
(c) ASC statistics with $\phi(t) = t\log(t)$ (KL)



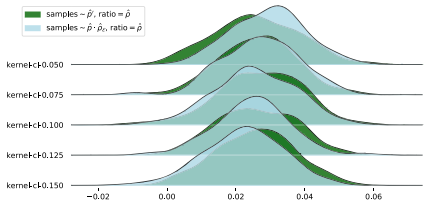
(d) ASC statistics with $\phi(t) = (\sqrt{t} - 1)^2$ (Hellinger)

Figure 10: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ with $\lambda = 0.8$ and different m, N, R, complementary to Fig. 4a. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

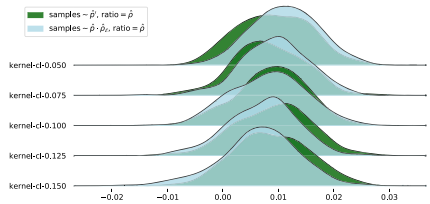
Question 2 (Fast Deletion). We visualize distributions of LR and ASC statistics between Y_{H_1} and $Y_{\mathcal{D}}$ in Fig. 11 (extension of Fig. 5a). The more overlapping between the distributions, the less distinguishable between the approximated and re-trained models. KBC is generally better than kNN. For kNN a moderate k (e.g. between 10 and 50) has better overlapping.



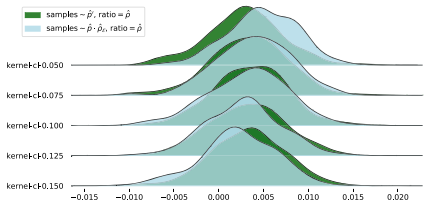
(a) LR for KBC-based DRE ($\lambda = 0.6$)



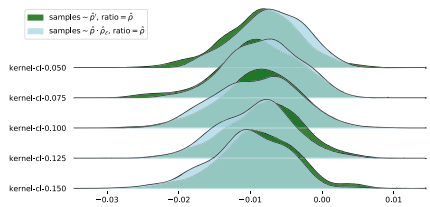
(b) LR for KBC-based DRE ($\lambda = 0.7$)



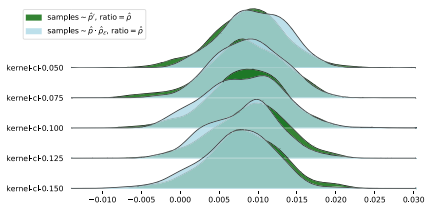
(c) LR for KBC-based DRE ($\lambda = 0.8$)



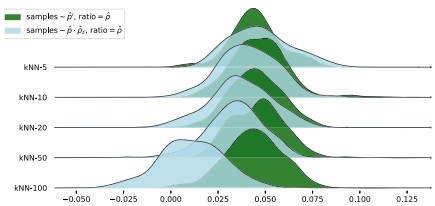
(d) LR for KBC-based DRE ($\lambda = 0.9$)



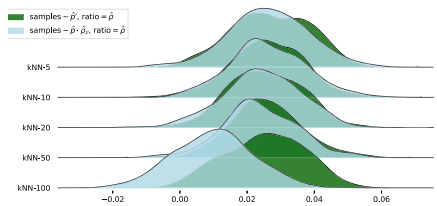
(e) ASC for KBC-based DRE ($\phi(t) = \log(t)$)



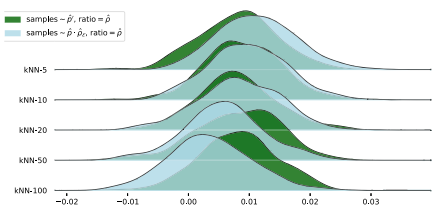
(f) ASC for KBC-based DRE ($\phi(t) = t\log(t)$)



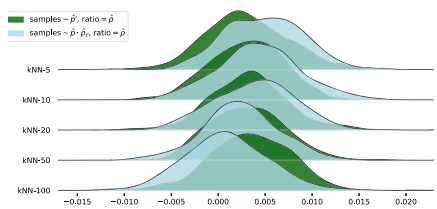
(g) LR for kNN-based DRE ($\lambda = 0.6$)



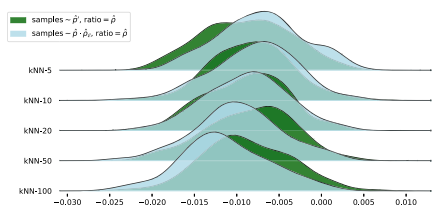
(h) LR for kNN-based DRE ($\lambda = 0.7$)



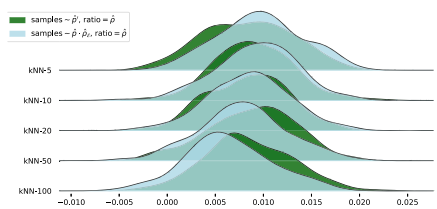
(i) LR for kNN-based DRE ($\lambda = 0.8$)



(j) LR for kNN-based DRE ($\lambda = 0.9$)



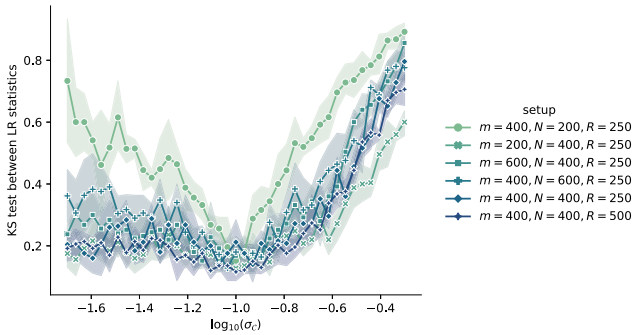
(k) ASC for kNN-based DRE ($\phi(t) = \log(t)$)



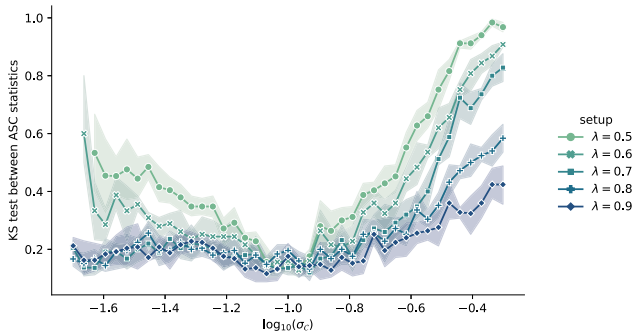
(l) ASC for kNN-based DRE ($\phi(t) = t\log(t)$)

Figure 11: (a)-(f) KBC-based DRE. (g)-(l) kNN-based DRE. (a)-(d)&(g)-(j) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$. (e)-(f)&(k)-(l) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{\mathcal{D}}, \hat{\rho})$.

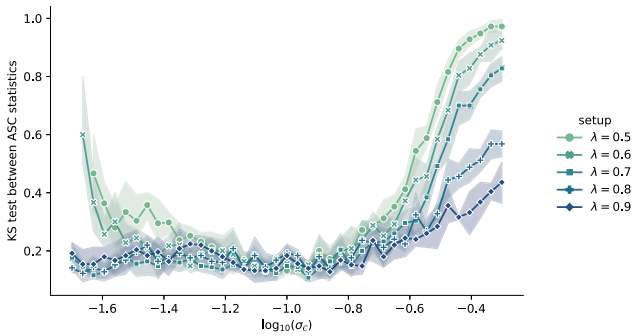
We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 12 (extension of Fig. 4b). The KS values are small for a wide range of $\sigma_{\mathcal{C}}$, indicating KBC with these $\sigma_{\mathcal{C}}$ can lead to approximated models indistinguishable from the re-trained model. There is no clear difference between LR and ASC statistics. In terms of λ, the models are less distinguishable when λ is larger, as expected.



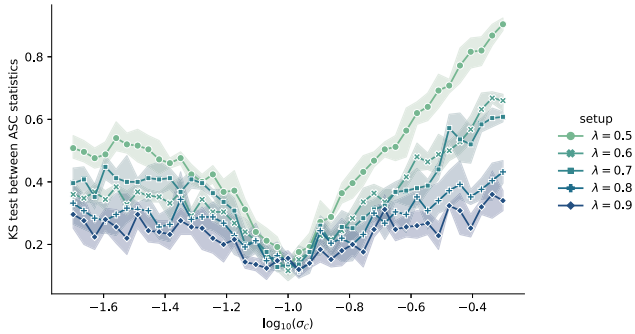
(a) LR statistics with $\lambda = 0.8$ and different m, N, R



(b) ASC statistics with $\phi(t) = \log(t)$



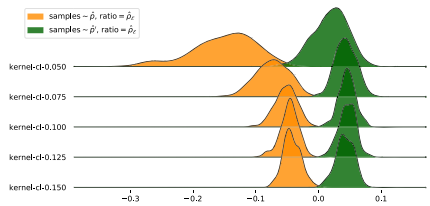
(c) ASC statistics with $\phi(t) = t\log(t)$



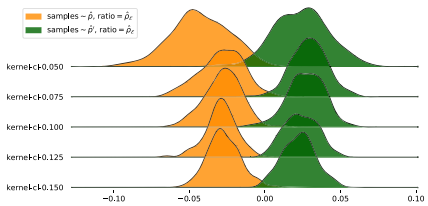
(d) ASC statistics with $\phi(t) = (\sqrt{t} - 1)^2$

Figure 12: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$ with $\lambda = 0.8$ and different m, N, R, complementary to Fig. 4b. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_1}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{\mathcal{D}}, \hat{\rho})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

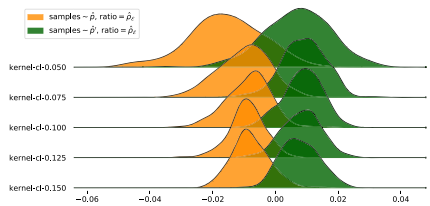
Question 3 (Hypothesis Test). We visualize distributions of LR and ASC statistics between Y_{H_0} and Y_{H_1} in Fig. 13 (extension of Fig. 5b). The separation between the distributions indicates how the DRE can distinguish samples between pre-trained and re-trained models. We observe separation for a wide range of classifiers, and KBC is generally comparable to kNN. In terms of statistics, the LR is better than ASC. In terms of λ, larger λ makes the two models less distinguishable.



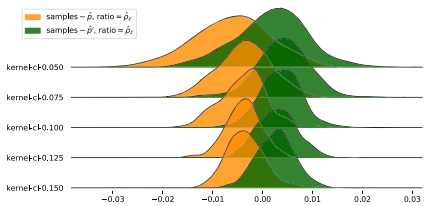
(a) LR for KBC-based DRE ($\lambda = 0.6$)



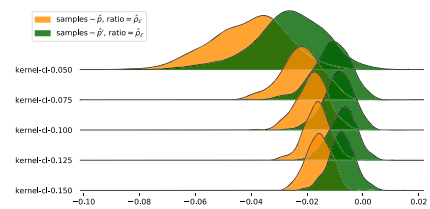
(b) LR for KBC-based DRE ($\lambda = 0.7$)



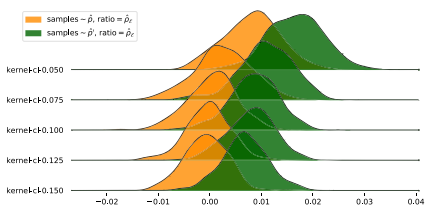
(c) LR for KBC-based DRE ($\lambda = 0.8$)



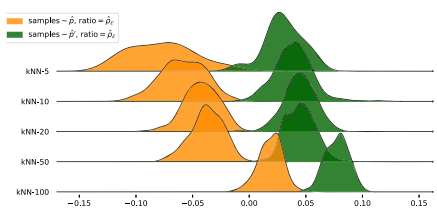
(d) LR for KBC-based DRE ($\lambda = 0.9$)



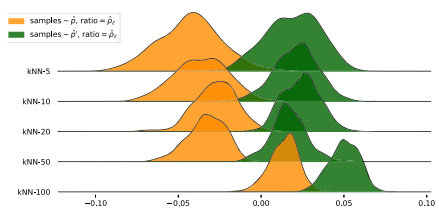
(e) ASC for KBC-based DRE ($\phi(t) = \log(t)$)



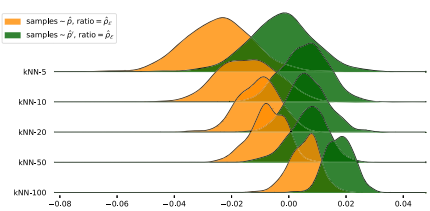
(f) ASC for KBC-based DRE ($\phi(t) = t\log(t)$)



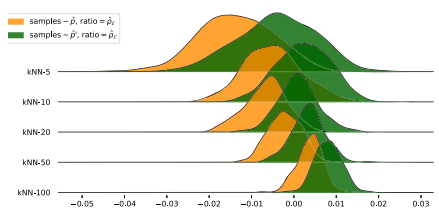
(g) LR for kNN-based DRE ($\lambda = 0.6$)



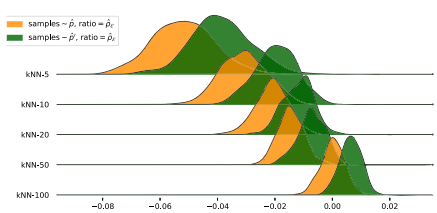
(h) LR for kNN-based DRE ($\lambda = 0.7$)



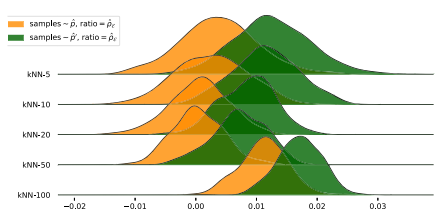
(i) LR for kNN-based DRE ($\lambda = 0.8$)



(j) LR for kNN-based DRE ($\lambda = 0.9$)



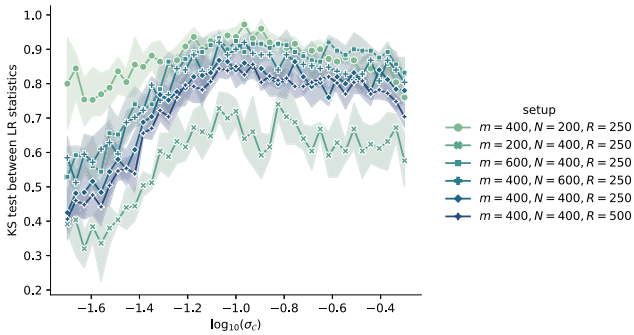
(k) ASC for kNN-based DRE ($\phi(t) = \log(t)$)



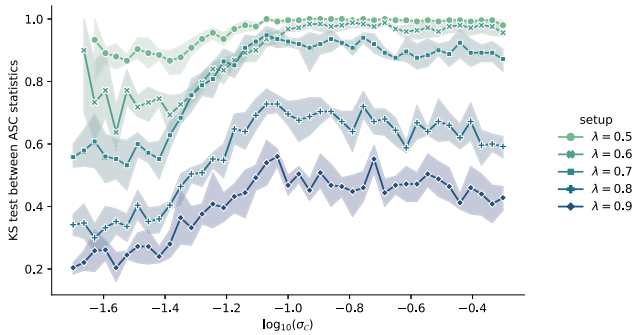
(l) ASC for kNN-based DRE ($\phi(t) = t\log(t)$)

Figure 13: (a)-(f) KBC-based DRE. (g)-(l) kNN-based DRE. (a)-(d)&(g)-(j) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$. (e)-(f)&(k)-(l) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$.

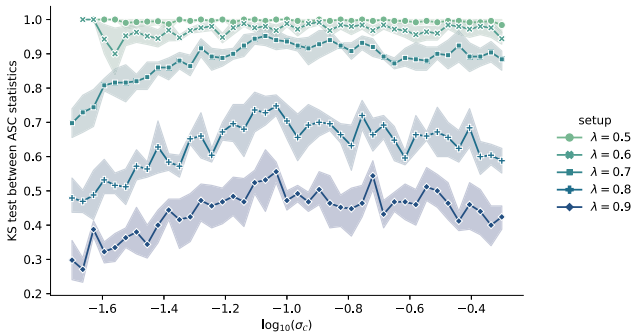
We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 14 (extension of Fig. 4c). The KS values are large for a wide range of $\sigma_{\mathcal{C}}$, indicating KBC with these $\sigma_{\mathcal{C}}$ can nicely distinguish pre-trained and re-trained model. LR statistics are slightly better than ASC statistics. In terms of λ, the models can be more easily distinguished when λ is small, as expected.



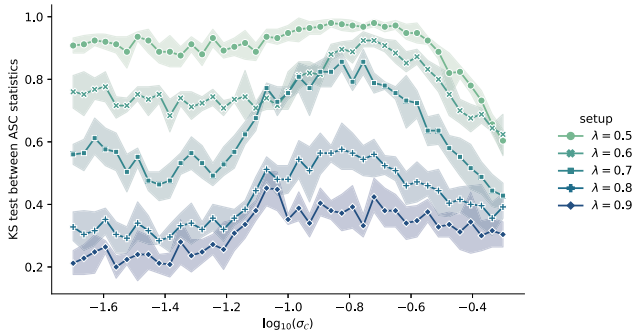
(a) LR statistics with $\lambda = 0.8$ and different m, N, R



(b) ASC statistics with $\phi(t) = \log(t)$



(c) ASC statistics with $\phi(t) = t\log(t)$



(d) ASC statistics with $\phi(t) = (\sqrt{t} - 1)^2$

Figure 14: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ with $\lambda = 0.8$ and different m, N, R, complementary to Fig. 4c. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_1}, \hat{\rho})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

D.2 CKB-8

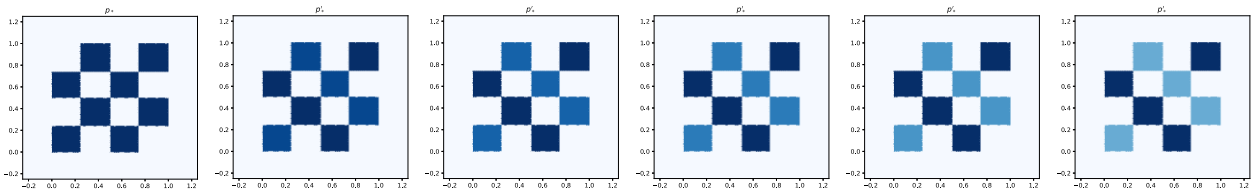
Setup. The CKB-8 data distribution is defined as

$$p_* = \texttt{Uniform}(\cup_{i=1}^{8}\Omega_i),$$

where $\Omega_1 = [0, 0.25] \times [0, 0.25], \Omega_2 = [0, 0.25] \times [0.5, 0.75], \Omega_3 = [0.25, 0.5] \times [0.25, 0.5], \Omega_4 = [0.25, 0.5] \times [0.75, 1], \Omega_5 = [0.5, 0.75] \times [0, 0.25], \Omega_6 = [0.5, 0.75] \times [0.5, 0.75], \Omega_7 = [0.75, 1] \times [0.25, 0.5], \Omega_8 = [0.75, 1] \times [0.75, 1]$. The modified distribution p'_* with weight λ is defined as

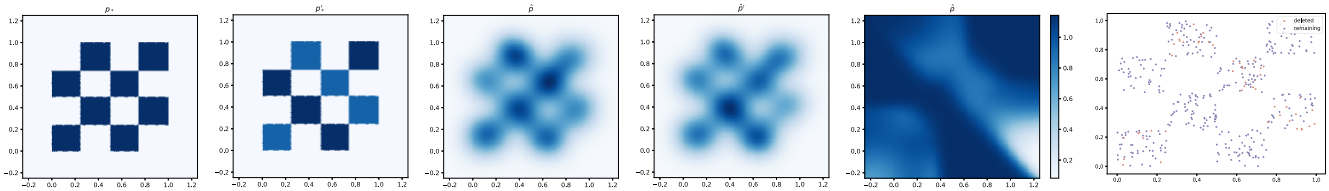
$$p'_*(x) = \frac{1}{4(1+\lambda)} \sum_{i=1}^{8} w_i \cdot \texttt{Uniform}(\Omega_i),$$

where $w_i = 1$ for $i \in \{2, 3, 5, 8\}$ and λ for $i \in \{1, 4, 6, 7\}$. The construction algorithm for X is randomly sampling a square id between 1 and 8 and randomly drawing a sample from the corresponding uniform distribution. The construction algorithm for X' is to include a sample $x \in X$ with probability $1 - \lambda$ if x is from i-th square for $i \in \{1, 4, 6, 7\}$. The distributions and data with different λ are shown in Fig. 15 and 16.



(a) p_* (b) $p'_*(\lambda = 0.9)$ (c) $p'_*(\lambda = 0.8)$ (d) $p'_*(\lambda = 0.7)$ (e) $p'_*(\lambda = 0.6)$ (f) $p'_*(\lambda = 0.5)$

Figure 15: Visualization of the experimental setup of CKB-8. (a) Data distribution p_*. (b) - (f) p'_* with different λ values. A larger λ means less data is deleted.



(a) p_* (b) p'_* (c) $\hat{p}$ (d) $\hat{p}'$ (e) $\hat{\rho}$ (f) X' and $X \setminus X'$

Figure 16: Visualization of the experimental setup of CKB-8. (a) Data distribution p_*. (b) Distribution p'_* with $\lambda = 0.8$. (c) Pre-trained KDE $\hat{p}$ on X with $\sigma_{\mathcal{A}} = 0.1$. (d) Re-trained KDE $\hat{p}'$ on $X \setminus X'$ with $\sigma_{\mathcal{A}} = 0.1$. (e) Density ratio $\hat{\rho} = \hat{p}'/\hat{p}$. (f) Deletion set X' and the remaining set $X \setminus X'$.

Other hyperparameters are set as follows. The number of training samples $N = 400$ unless specified. The number of samples for the deletion test $m = 400$ unless specified. The number of repeats for each setup is $R = 250$ unless specified. The learning algorithm KDE has bandwidth $\sigma_{\mathcal{A}} = 0.1$ unless specified.

Question 1 (DRE Approximations). We visualize $\hat{\rho}$ and $\hat{\rho}_{\mathcal{E}}$ in Fig. 17 (extension of Fig. 3 for CKB-8). These figures give qualitative answers to question 1 (DRE Approximations).

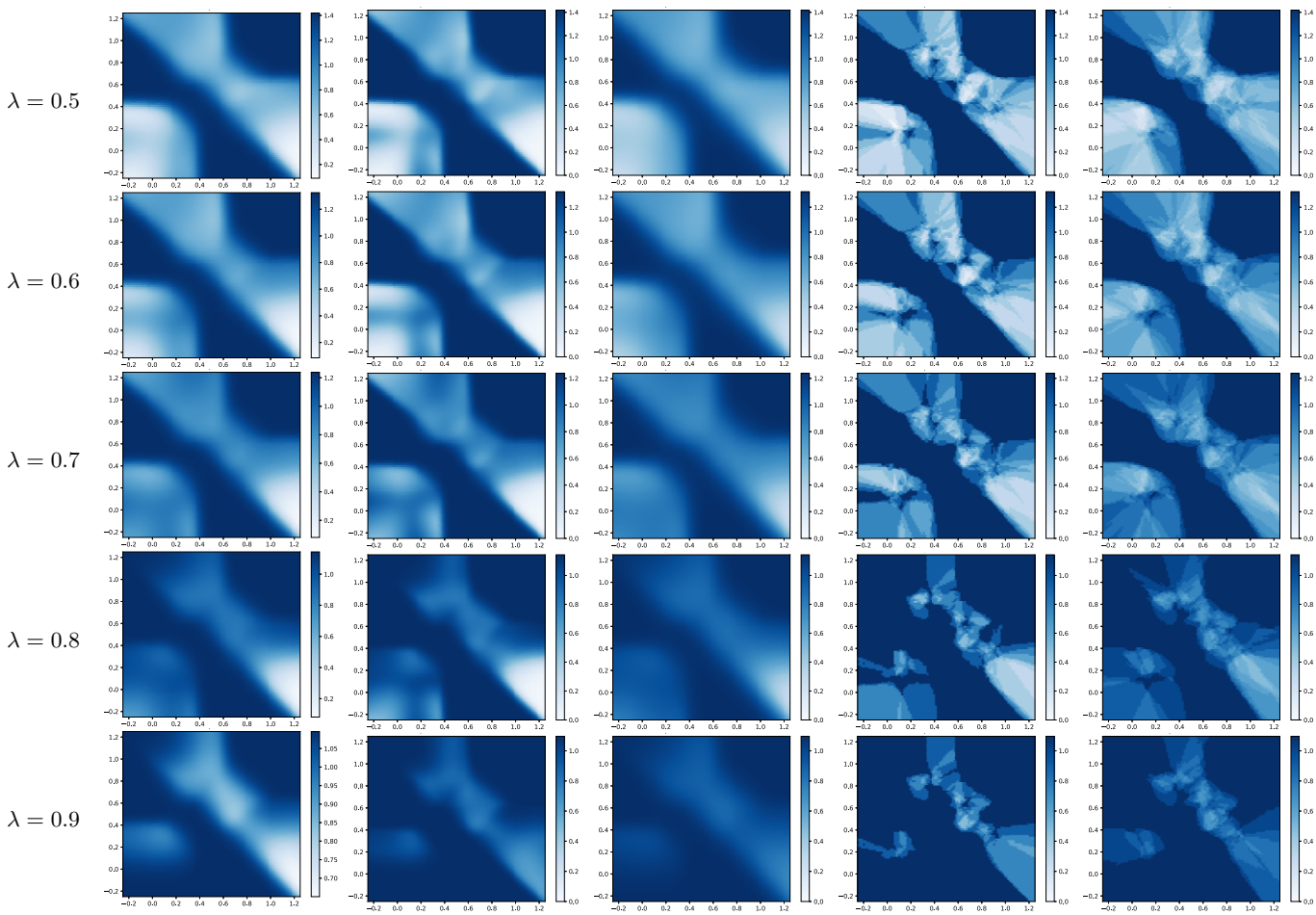
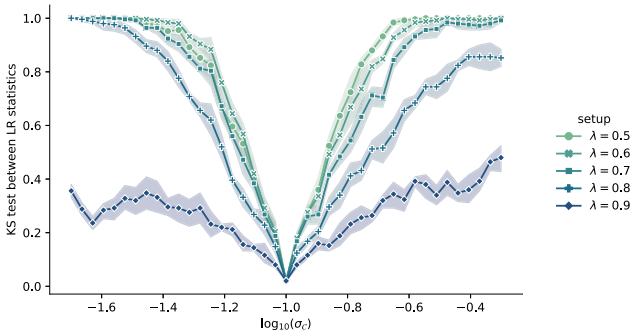
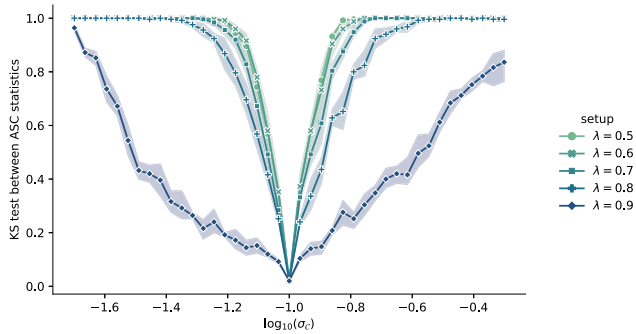


Figure 17: Visualization of ratio $\hat{\rho}$ in (a) and $\hat{\rho}_{\mathcal{E}}$ in (b)-(e) for different classifier-based DREs.

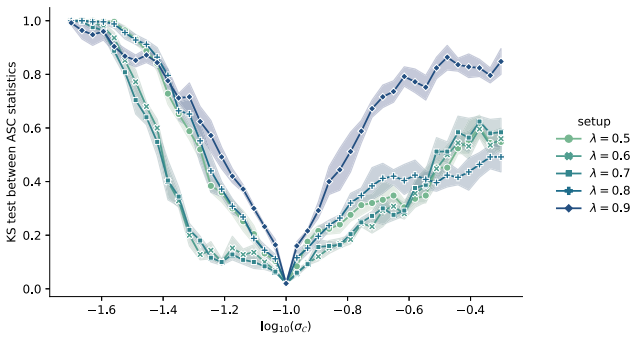
We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 18 (extension of Fig. 4a for CKB-8). When $\sigma_{\mathcal{C}} \approx \sigma_{\mathcal{A}} = 0.1$, the KS values are small, indicating KBC with these $\sigma_{\mathcal{C}}$ can lead to classifier-based DRE $\hat{\rho}_{\mathcal{E}}$ that is close to $\hat{\rho}$. Comparing to MoG-8, the conclusion for CKB-8 is similar, but estimation is harder. One exception is when $\lambda = 0.9$, the estimation is very accurate under LR or ASC with $\phi(t) = \log(t)$, but not as accurate as other λ under ASC with $\phi(t) = t\log(t)$.



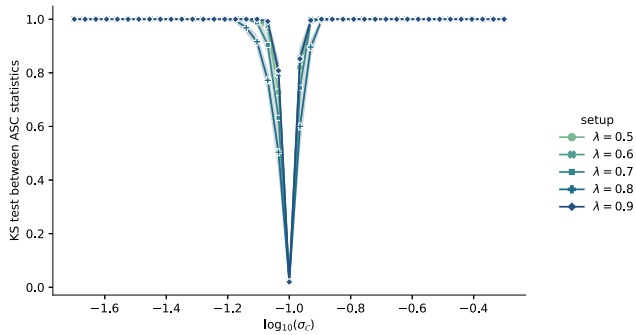
(a) LR statistics



(b) ASC statistics with $\phi(t) = \log(t)$



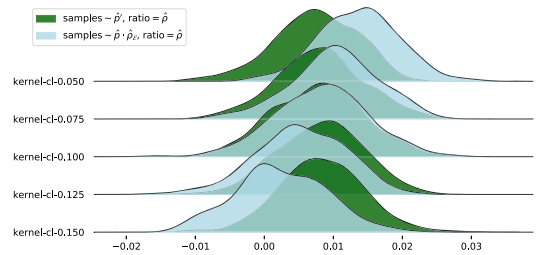
(c) ASC statistics with $\phi(t) = t\log(t)$ (KL)



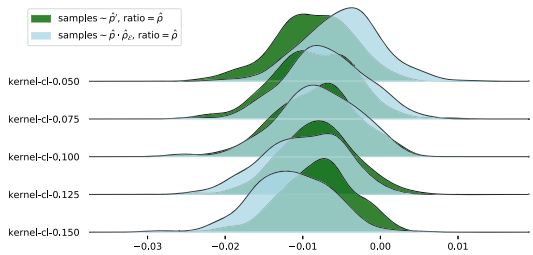
(d) ASC statistics with $\phi(t) = (\sqrt{t} - 1)^2$ (Hellinger)

Figure 18: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ with different λ. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho}_{\mathcal{E}})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

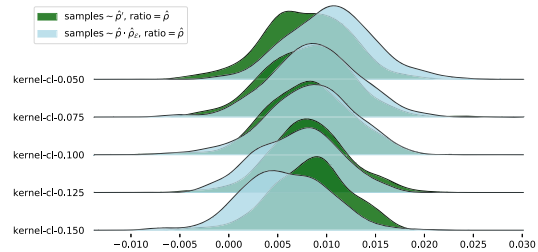
Question 2 (Fast Deletion). We visualize distributions of LR and ASC statistics between Y_{H_1} and $Y_{\mathcal{D}}$ in Fig. 19 (extension of Fig. 5a for CKB-8). The more overlapping between the distributions, the less distinguishable between the approximated and re-trained models. For both KBC and kNN, the distribution pairs are slightly more separated than MoG-8, indicating fast deletion is harder for CKB-8. For kNN a moderate k (e.g. between 10 and 50) has better overlapping.



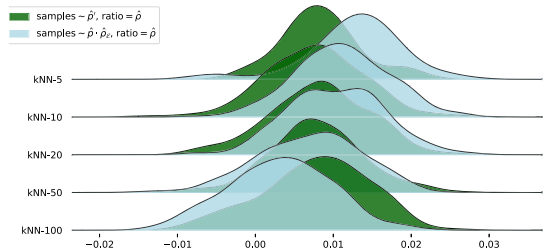
(a) LR for KBC-based DRE ($\lambda = 0.8$)



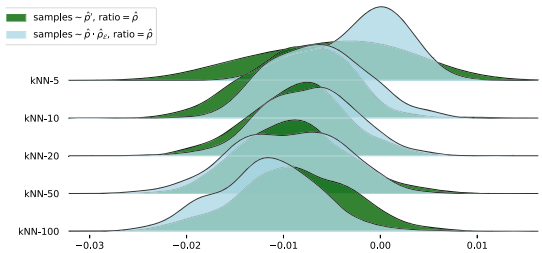
(b) ASC for KBC-based DRE ($\phi(t) = \log(t)$)



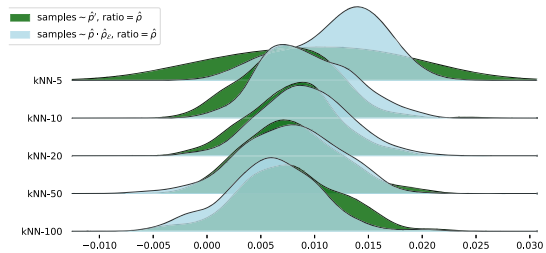
(c) ASC for KBC-based DRE ($\phi(t) = t\log(t)$)



(d) LR for kNN-based DRE ($\lambda = 0.8$)



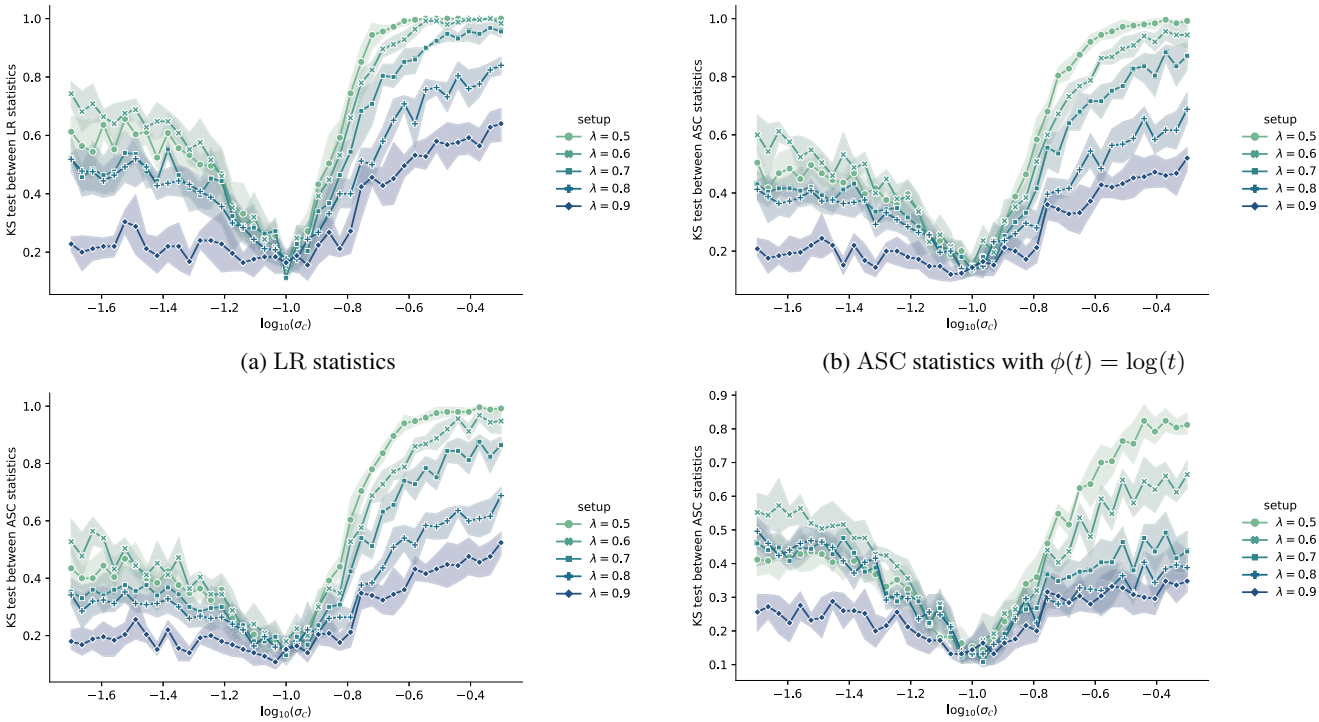
(e) ASC for kNN-based DRE ($\phi(t) = \log(t)$)



(f) ASC for kNN-based DRE ($\phi(t) = t\log(t)$)

Figure 19: (a)-(c) KBC-based DRE. (d)-(f) kNN-based DRE. (a)&(d) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$. (b)-(c)&(e)-(f) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{\mathcal{D}}, \hat{\rho})$.

We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 20 (extension of Fig. 4b for CKB-8). The conclusions for CKB-8 are similar to MoG-8, except that the KS values are slightly higher, indicating the fast deletion is slightly harder for this dataset.



(a) LR statistics

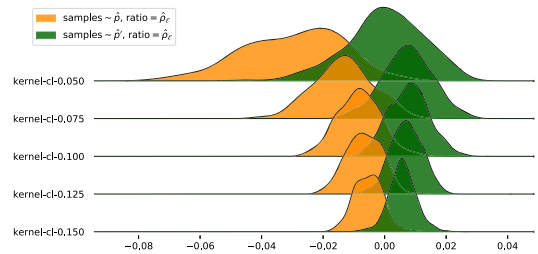
(b) ASC statistics with $\phi(t) = \log(t)$

(c) ASC statistics with $\phi(t) = t\log(t)$

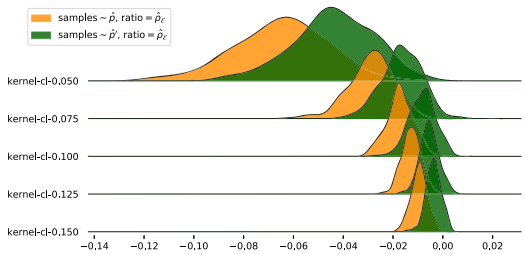
(d) ASC statistics with $\phi(t) = (\sqrt{t} - 1)^2$

Figure 20: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ vs $\mathrm{LR}(Y_{\mathcal{D}}, \hat{\rho})$ with $\lambda = 0.8$. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_1}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{\mathcal{D}}, \hat{\rho})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

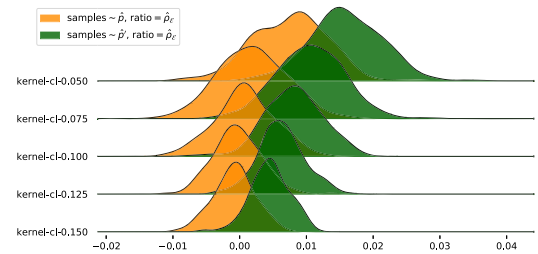
Question 3 (Hypothesis Test). We visualize distributions of LR and ASC statistics between Y_{H_0} and Y_{H_1} in Fig. 21 (extension of Fig. 5b for CKB-8). The separation between the distributions indicates how the DRE can distinguish samples between pre-trained and re-trained models. Similar to MoG-8, LR statistics lead to better separation than ASC.



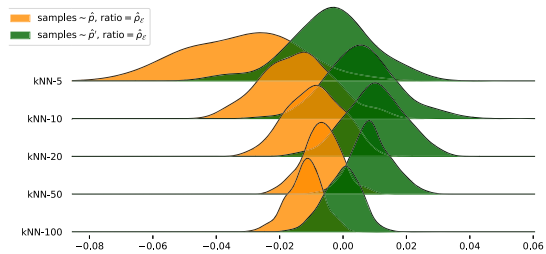
(a) LR for KBC-based DRE ($\lambda = 0.8$)



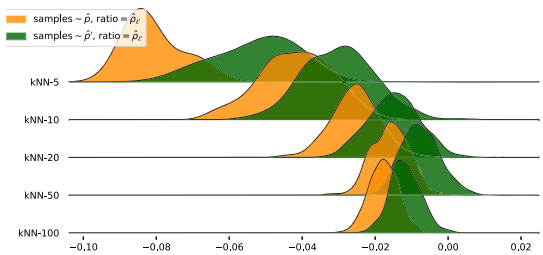
(b) ASC for KBC-based DRE ($\phi(t) = \log(t)$)



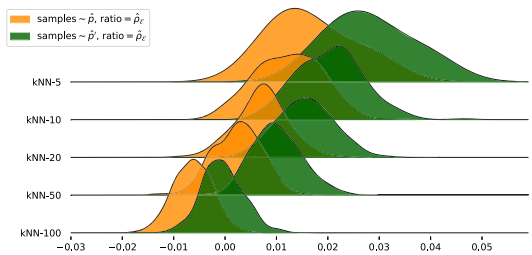
(c) ASC for KBC-based DRE ($\phi(t) = t\log(t)$)



(d) LR for kNN-based DRE ($\lambda = 0.8$)



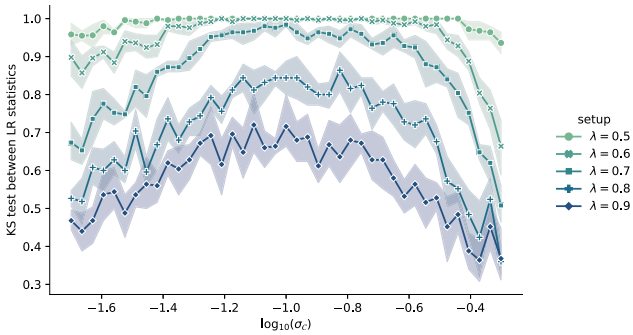
(e) ASC for kNN-based DRE ($\phi(t) = \log(t)$)



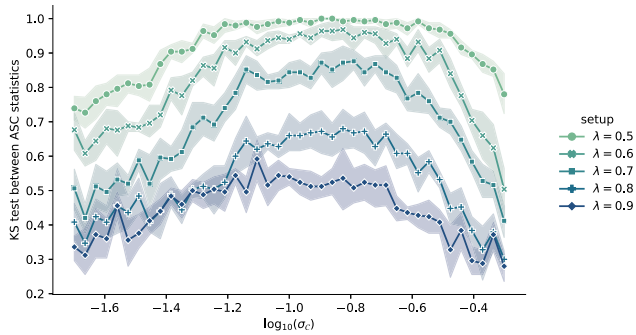
(f) ASC for kNN-based DRE ($\phi(t) = t\log(t)$)

Figure 21: (a)-(c) KBC-based DRE. (d)-(f) kNN-based DRE. (a)&(d) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$. (b)-(c)&(e)-(f) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$.

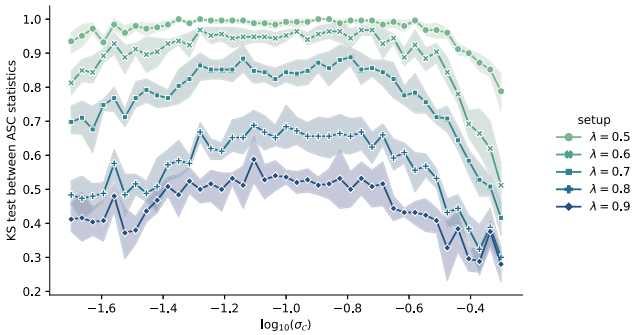
We visualize KS test results for KBC with different bandwidth $\sigma_{\mathcal{C}}$ in Fig. 22 (extension of Fig. 4c for CKB-8). Similar to MoG-8, LR statistics lead to better separation than ASC.



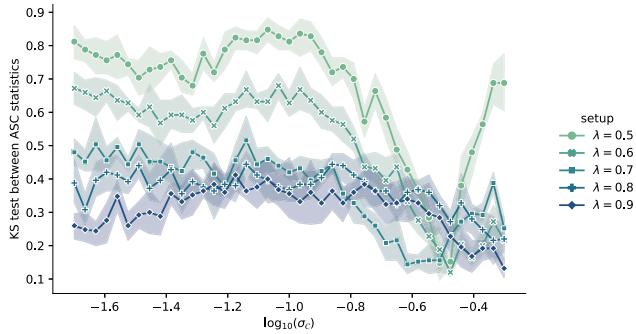
(a) LR statistics



(b) ASC statistics with $\phi(t) = \log(t)$



(c) ASC statistics with $\phi(t) = t\log(t)$



(d) ASC statistics with $\phi(t) = (\sqrt{t}-1)^2$

Figure 22: KS tests between distributions of statistics for KBC with different $\sigma_{\mathcal{C}}$. (a) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$ with $\lambda = 0.8$. (b)-(d) $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_{\phi}(\hat{Y}, Y_{H_1}, \hat{\rho})$ for different ϕ. Smaller values indicate the two compared distributions are closer.

E Experiments on GAN

E.1 Setup

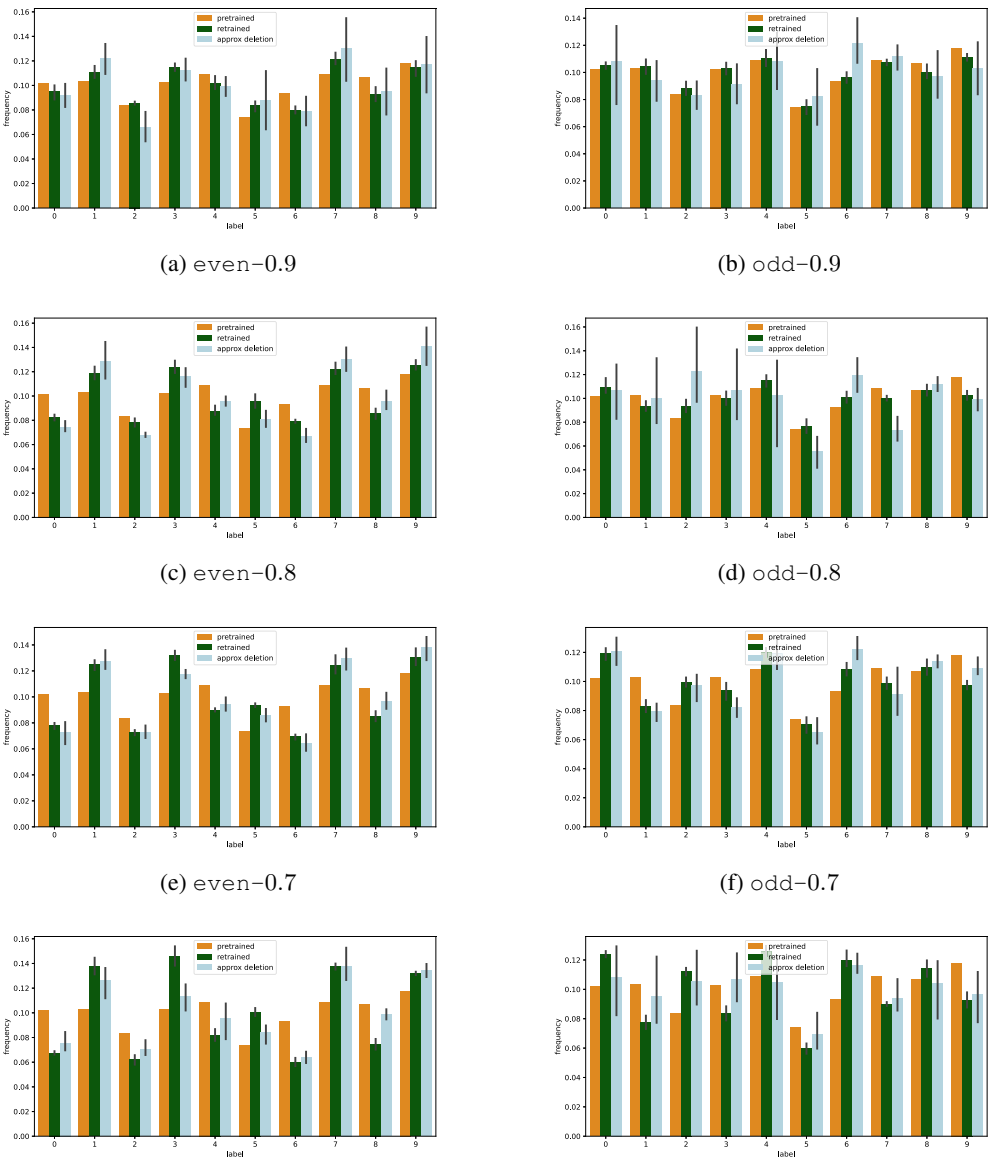
We run experiments on MNIST [26] and Fashion-MNIST [46]. Both datasets contain gray-scale 28×28 images with 10 labels $\{0, 1, \cdots, 9\}$. We define the $\texttt{even-}\lambda$ setting as the subset containing all samples with odd labels and a λ fraction of samples with even labels randomly selected from the training set. The rest $1 - \lambda$ fraction of samples with even labels form the deletion set X'. We have similar definition for $\texttt{odd-}\lambda$. In experiments, we let $\lambda \in \{0.6, 0.7, 0.8, 0.9\}$.

The learner is a DCGAN [36]. For pre-trained and re-trained models, we train each of them for 200 epochs. To obtain DRE, we optimize (5), where the network T has the same architecture as the discriminator and is trained for 40 epochs. The learning rate is halved for stability.

All experiments were run on a single machine with one i9-9940X CPU (3.30GHz), one 2080Ti GPU, and 128GB memory.

E.2 Results on MNIST

Question 2 (Fast Deletion). We generate $m = 50$K samples from pre-retrained, re-trained, and approximated models (with rejection sampling bound $B = 10$). We then compute the label distributions of these samples based on pre-trained classifiers. [7] Results for each deletion set (including means and standard errors for five random seeds) are shown in Fig. 23. We find the approximated model generates less (even or odd) labels some data with these labels are deleted from the training set. The variances for deleting odd labels are higher than deleting even labels.



(a) even−0.9 (b) odd−0.9

(c) even−0.8 (d) odd−0.8

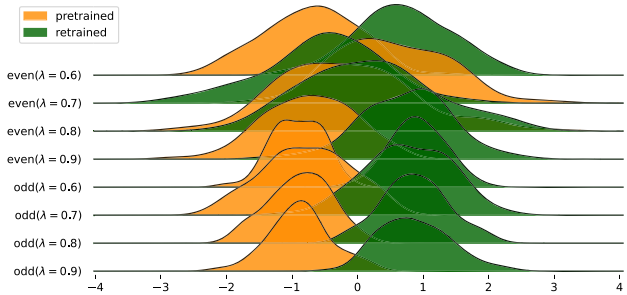
(e) even−0.7 (f) odd−0.7

(g) even−0.6 (h) odd−0.6

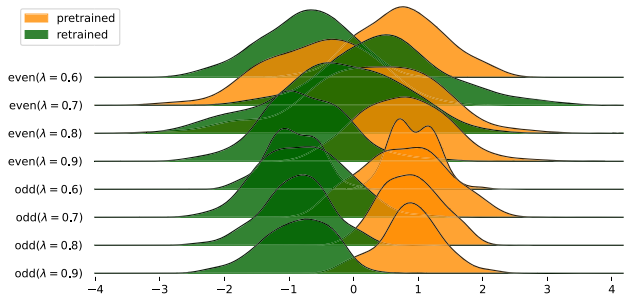
Figure 23: Label distributions of samples from pre-trained, re-trained, and approximated models. The closeness between green and light blue distributions indicate how well the fast deletion performs.

[7] https://github.com/aaron-xichen/pytorch-playground (MIT license)

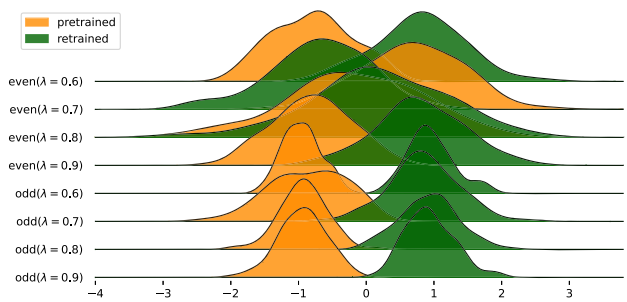
Question 3 (Hypothesis Test). We generate $m = 1000$ samples for each Y_{H_i}, $i = 1, 2$, and $\hat{Y}$. We visualize distributions of LR and ASC statistics between Y_{H_0} and Y_{H_1} in Fig. 24. The separation between the distributions indicates how the DRE can distinguish samples between pre-trained and re-trained models. The separation for `odd-`λ is better than `even-`λ. In terms of statistics, the LR is slightly better than ASC. In terms of λ, a smaller λ does not lead to more separation.



(a) ASC for VDM-based DRE ($\phi(t) = \log(t)$)



(b) ASC for VDM-based DRE ($\phi(t) = t\log(t)$)

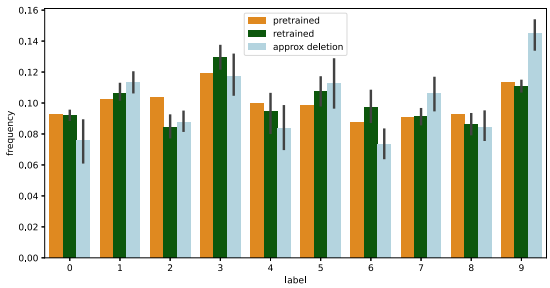


(c) LR for VDM-based DRE

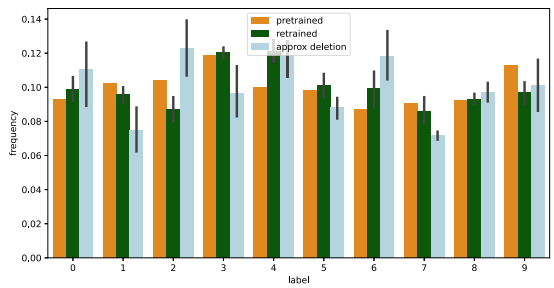
Figure 24: (a)-(b) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$. (c) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$

E.3 Results on Fashion-MNIST

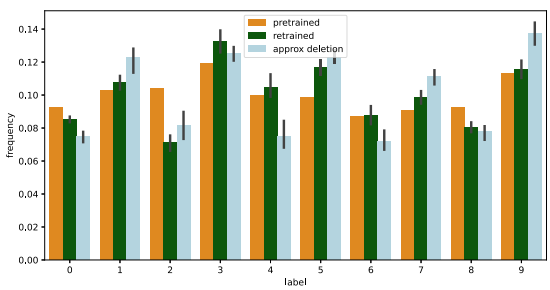
Question 2 (Fast Deletion). Label distributions for each deletion set (including means and standard errors for five random seeds) are shown in Fig. 25. Similar to MNIST, we find the approximated model generates less (even or odd) labels some data with these labels are deleted from the training set., and the variances for deleting odd labels are slightly higher than deleting even labels.



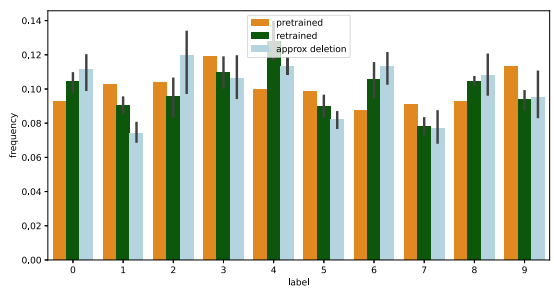
(a) even−0.9



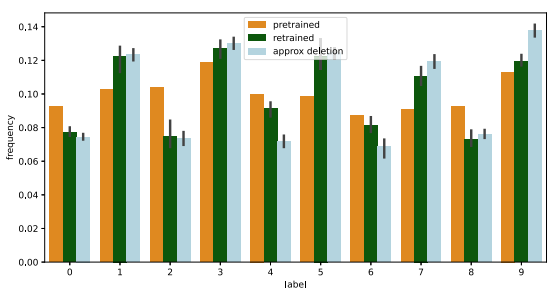
(b) odd−0.9



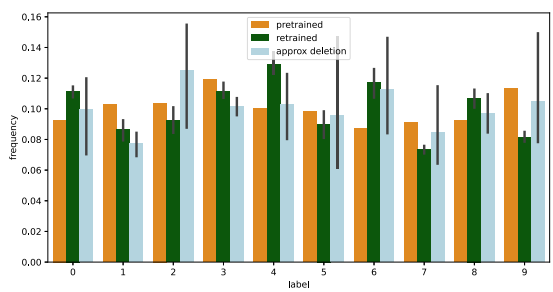
(c) even−0.8



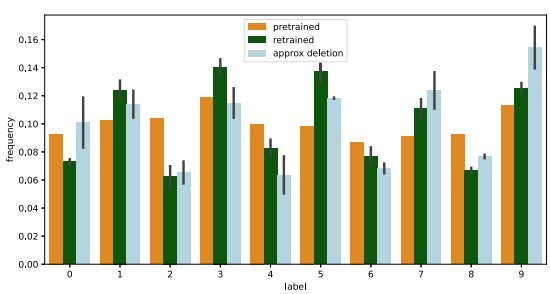
(d) odd−0.8



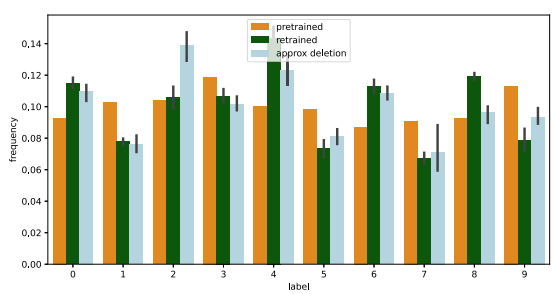
(e) even−0.7



(f) odd−0.7



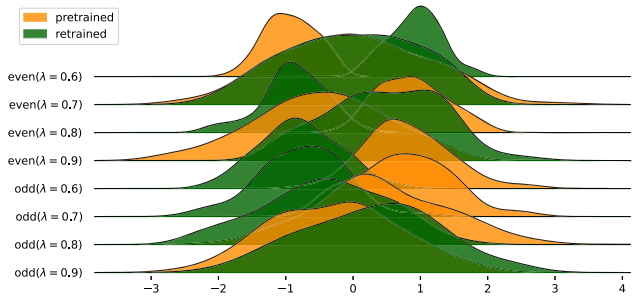
(g) even−0.6



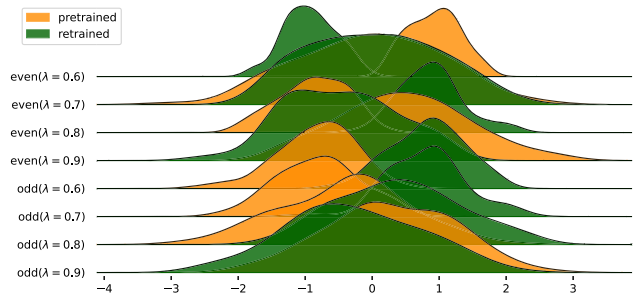
(h) odd−0.6

Figure 25: Label distributions of samples from pre-trained, re-trained, and approximated models. The closeness between green and light blue distributions indicate how well the fast deletion performs.

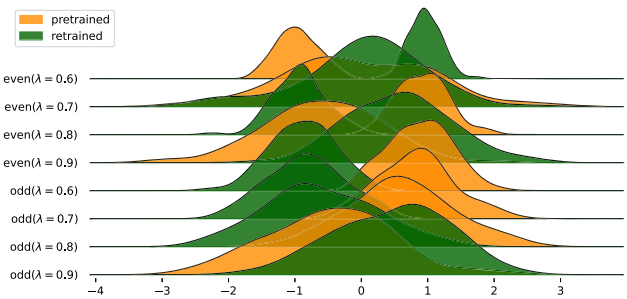
Question 3 (Hypothesis Test). We generate $m = 1000$ samples for each Y_{H_i}, $i = 1, 2$, and $\hat{Y}$. We visualize distributions of LR and ASC statistics between Y_{H_0} and Y_{H_1} in Fig. 26. The separation between the distributions indicates how the DRE can distinguish samples between pre-trained and re-trained models. The separation is good for some deletion sets (e.g. $\lambda = 0.6$) while not obvious for others (e.g. $\lambda = 0.9$), indicating performing the deletion test for Fashion-MNIST is harder than MNIST. There is no significant differences between LR and ASC statistics.



(a) ASC for VDM-based DRE ($\phi(t) = \log(t)$)



(b) ASC for VDM-based DRE ($\phi(t) = t\log(t)$)



(c) LR for VDM-based DRE

Figure 26: (a)-(b) $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_0}, \hat{\rho})$ vs $\hat{\mathrm{ASC}}_\phi(\hat{Y}, Y_{H_1}, \hat{\rho})$. (c) $\mathrm{LR}(Y_{H_0}, \hat{\rho})$ vs $\mathrm{LR}(Y_{H_1}, \hat{\rho})$

E.4 Comparison of Generation Quality

We measure the Inception Scores (IS) of the pre-trained, re-trained, and approximate deletion model. The results for MNIST are shown in Table 4, and the results for Fashion-MNIST are shown in Table 5. The generation qualities are highly comparable to each other.

Deleted set	IS (pre-trained)	IS (re-trained)	IS (approximate deletion)
even-0.9	7.66 ± 0.01	7.61 ± 0.05	7.56 ± 0.19
odd-0.9	7.66 ± 0.01	7.68 ± 0.10	7.64 ± 0.34
even-0.8	7.66 ± 0.01	7.60 ± 0.09	7.49 ± 0.05
odd-0.8	7.66 ± 0.00	7.67 ± 0.02	7.59 ± 0.20
even-0.7	7.66 ± 0.01	7.52 ± 0.08	7.47 ± 0.05
odd-0.7	7.66 ± 0.01	7.68 ± 0.07	7.65 ± 0.08
even-0.6	7.66 ± 0.01	7.29 ± 0.04	7.52 ± 0.13
odd-0.6	7.66 ± 0.01	7.60 ± 0.06	7.11 ± 0.83

Table 4: Inception Score Comparison (MNIST).

Deleted set	IS (pre-trained)	IS (re-trained)	IS (approximate deletion)
even-0.9	6.62 ± 0.01	6.57 ± 0.16	6.69 ± 0.10
odd-0.9	6.62 ± 0.01	6.48 ± 0.06	6.28 ± 0.12
even-0.8	6.62 ± 0.01	6.55 ± 0.08	6.75 ± 0.06
odd-0.8	6.62 ± 0.01	6.40 ± 0.10	6.22 ± 0.14
even-0.7	6.62 ± 0.01	6.73 ± 0.08	6.76 ± 0.03
odd-0.7	6.62 ± 0.01	6.34 ± 0.13	6.27 ± 0.13
even-0.6	6.62 ± 0.01	6.45 ± 0.04	6.69 ± 0.08
odd-0.6	6.62 ± 0.01	6.14 ± 0.15	6.29 ± 0.08

Table 5: Inception Score Comparison (Fashion-MNIST).