General-Purpose f-DP Estimation and Auditing in a Black-Box Setting

Önder Askin[1], Holger Dette[1], Martin Dunsche[1,*] Tim Kutta[2], Yun Lu[3], Yu Wei[4], Vassilis Zikas[4†]

[1] *Ruhr-University Bochum*
[2] *Aarhus University*
[3] *University of Victoria*
[4] *Georgia Institute of Technology*

Abstract

In this paper we propose new methods to statistically assess f-Differential Privacy (f-DP), a recent refinement of differential privacy (DP) that remedies certain weaknesses of standard DP (including tightness under algorithmic composition). A challenge when deploying differentially private mechanisms is that DP is hard to validate, especially in the black-box setting. This has led to numerous empirical methods for auditing standard DP, while f-DP remains less explored. We introduce new black-box methods for f-DP that, unlike existing approaches for this privacy notion, do not require prior knowledge of the investigated algorithm. Our procedure yields a complete estimate of the f-DP trade-off curve, with theoretical guarantees of convergence. Additionally, we propose an efficient auditing method that empirically detects f-DP violations with statistical certainty, merging techniques from non-parametric estimation and optimal classification theory. Through experiments on a range of DP mechanisms, we demonstrate the effectiveness of our estimation and auditing procedures.

1 Introduction

Differential privacy (DP) [20] is a widely-used framework to quantify and limit information leakage of a data-release *mechanism* M via privacy parameters $\varepsilon > 0$ and $\delta \in [0,1]$. Mechanisms that are differentially private for a suitable choice of ε and δ mask the contribution of individuals to their output. As a consequence, DP has been adopted by companies and public institutions to ensure user privacy [1, 21, 24].

Over the years, variants and relaxations of DP have been proposed to address specific needs and challenges. Of these, the recent notion of f-DP [19] is one of the most notable, due to its attractive properties such as a tight composition theorem, and applications such as providing an improved, simpler analysis of privatized stochastic gradient descent (Noisy or DP-SGD), the most prominent privacy-preserving algorithm in machine learning. f-DP is grounded on the hypothesis testing interpretation of DP [1] and describes the privacy of mechanism M in terms of a real-valued function f on the unit interval $[0,1]$. Several mechanisms [19] have been shown to achieve f-DP. However, the process of designing privacy-preserving mechanisms and turning them into real-world implementations is susceptible to errors that can lead to so-called 'privacy violations' [27, 35, 37]. Worse, checking such claims may be difficult, as some implementations may only allow for limited, *black-box* access. This problem has motivated the proposal of methods that assess the privacy of a mechanism M with only black-box access.

Within the plethora of works on privacy validation, most approaches study mechanisms through the lens of standard DP [6–13, 16, 18, 32–34, 44, 48, 49, 52]. In contrast, comparatively few methods examine f-DP [3–5, 30, 36, 38]. Moreover, many of the procedures that feature f-DP are tailored to audit the privacy claims of a specific algorithm, namely DP-SGD [3, 4, 38]. Our goal is to devise methods that are not specific to a single mechanism, but are instead applicable to a broad class of algorithms, while only requiring black-box access. We formulate our two objectives:

- **Estimation:** Given black-box access to a mechanism M, estimate its true privacy parameter (i.e., the function f in f-DP).
- **Auditing:** Given black-box access to a mechanism M and a target privacy f, check whether M violates the targeted privacy level (i.e., given f, does M satisfy f-DP?).

Estimation is useful when we do not have an initial conjecture regarding M's privacy. It can thus be used as, e.g.,

[*] Corresponding author: `martin.dunsche@rub.de`
[†] Authors are listed in alphabetical order.

[1] For a rigorous introduction to hypothesis testing and f-DP we refer to Section 2.

preliminary exploration into the privacy of M. Auditing, on the other hand, can check whether an algorithm meets a specific target privacy f and is therefore designed to detect flaws or overly optimistic privacy guarantees. [2]

Contributions

We construct a 'general-purpose' f-DP *estimator* and *auditor* for both objectives, where:

(1) The estimator approximates the entire true f-DP curve of a given mechanism M.

(2) Given a target f-DP curve, the auditor statistically detects whether M violates f-DP. The auditor involves a tuneable confidence parameter to control the false detection rate.

A methodological advantage of our methods is that they come with strong mathematical performance guarantees (both for the estimator and the auditor). Such guarantees seem warranted when making claims about the performance and correctness of a mechanism. A practical advantage of our methods is their efficiency: Our experiments (Sec. 6) demonstrate high accuracy at typical runtimes of 1-2 minutes on a standard personal device.

Paper Organization Preliminaries are introduced in Sec. 2. In Sec. 3 we give an overview of techniques. We propose our f-DP curve estimator in Sec. 4 and auditor in Sec. 5. We evaluate the effectiveness of both estimator and auditor in Sec. 6 using various mechanisms from the DP literature, including DP-SGD. We delve into more detail on related work in Sec. 7 and conclude in Sec. 8. A table of notations, proofs and technical details can be found in the Appendix.

2 Preliminaries

In this section, we provide details on hypothesis testing, differential privacy and tools from statistics and machine learning that our methods rely on.

2.1 Hypothesis testing

We provide a brief introduction into the key concepts of hypothesis testing. We confine ourselves to the special case of sample size 1, most relevant to f-DP. For a general introduction we refer to [14]. Consider two probability distributions P,Q on the Euclidean space $\mathbb{R}^d$ and a random variable X. It is unknown from which of the two distributions X is drawn and the task is to decide between the two competing hypotheses

$$H_0 : X \sim P \quad \text{vs.} \quad H_1 : X \sim Q. \tag{1}$$

The problem is similar to a classification task (see Section 2.4 below). The key difference to classification is that, in hypothesis testing, there exists a default belief H_0 that is preferred over H_1. The user switches from H_0 to H_1 only if the data (X) suggests it strongly enough. In this context, a hypothesis test is a binary, potentially randomized function $g : \mathbb{R}^d \to \{0,1\}$, where $g(X) = 0$ implies to stay with H_0, while $g(X) = 1$ implies that the user should switch to H_1 (H_0 is "rejected"). Just as in classification, the decision to reject/fail to reject can be erroneous and the error rates of these decisions are called α, the "type-I error", and β, the "type-II error". Their formal definitions are

$$\alpha^{(g)} := \Pr_{X\sim P}[g(X) = 1], \quad \beta^{(g)} := \Pr_{X\sim Q}[g(X) = 0].$$

One test g is better than another g', if simultaneously

$$\alpha^{(g)} \le \alpha^{(g')} \quad \text{and} \quad \beta^{(g)} \le \beta^{(g')}.$$

This comparison of statistical tests naturally leads to the issue of optimal tests, and we define the optimal level-α-test as the argmin of

$$\{\beta^{(g)} : g \text{ is a test with } \alpha^{(g)} \le \alpha\}.$$

The minimum is achieved and the corresponding optimal test is provided by the *likelihood ratio (LR) test* in the Neyman-Pearson lemma, a fundamental result in statistics. In the following, we assume the two probability measures P,Q in hypotheses (1) have some probability densities p,q.

Theorem 2.1 (Neyman-Pearson Lemma [39]) *For any $\alpha \in [0,1]$, the smallest type-II error $\beta(\alpha)$ among all level-α-tests is achieved by the* **likelihood ratio (LR) test***, which is characterized by two constants $\eta \ge 0$ and $\lambda \in [0,1]$, and has the following rejection rule:*

1) Reject H_0 if $q(X)/p(X) > \eta$.

2) If $q(X)/p(X) = \eta$, flip an unfair coin with probability λ of heads. If the outcome is heads, reject H_0.

The constants (η,λ) are chosen such that the type-I error is exactly α.

Notations. Neyman-Pearson motivates the use of the following notations. For any type-I error α there is a corresponding (optimal) β implied by the lemma. These constants are achieved by a pair (η,λ) and we can thus write $\alpha(\eta,\lambda), \beta(\eta,\lambda)$ for them. When we are only interested in the result of the non-randomized test with $\lambda = 0$, we will just write $\alpha(\eta), \beta(\eta)$.

[2] For a detailed discussion on the advantages of auditing f-DP, we refer to Section 4 in [38].

2.2 (f-)Differential Privacy (DP)

DP requires that the output of mechanism M is similar on all *neighboring* datasets D, D' that differ in exactly one data point (we also call D, D' *neighbors*). We use the "edit" notion of neighborhood, i.e., D' can be obtained from D by editing one of its entries, rather than deleting it.

Definition 1 (DP [20]) *A mechanism M is (ε,δ)-DP if for all neighboring datasets D, D' and any set $\mathcal{S}$,*

$$\Pr(M(D) \in \mathcal{S}) \leq e^{\varepsilon} \Pr(M(D') \in \mathcal{S}) + \delta \, .$$

Informally, if M is (ε,δ)-DP, an adversary's ability to decide whether M was run on D or D' is bounded by δ and e^{ε}. For instance, any statistical level-α-test g that aims at deciding this problem must incur a type-II-error of at least $1 - e^{\varepsilon}\alpha - \delta$. The notion of f-DP was introduced to make this observation more rigorous. Given a pair of neighbors D and D' and a sample X, consider the hypotheses:

$$H_0 : X \sim P \qquad\qquad H_1 : X \sim Q,$$

where $M(D)$ and $M(D')$ are distributed to P, Q, respectively. Roughly speaking, good privacy requires these two hypotheses to be hard to distinguish. That is, for any hypothesis test with type-I error α, its type-II error β should be large. This is captured by the trade-off function T between P and Q.

Definition 2 (Trade-off function [19]) *For any two distributions P and Q on the same space, the trade-off function T is:*

$$T(\alpha) := \inf\{\beta^{(g)} : g \text{ test with } \alpha^{(g)} \leq \alpha\}$$

M is f-DP if its privacy is at least as good (its trade-off function is at least as large) as f, when considering all neighboring datasets.

Definition 3 (f-DP [19]) *A mechanism M is f-DP if for all neighboring datasets D, D' it holds that $T \geq f$. Here, T is the trade-off function implied by $M(D) \sim P$ and $M(D') \sim Q$.*

We say f is the *optimal/true* privacy parameter if it is the largest f such that M is f-DP—such optimality is necessary to define for meaningful f-DP estimation, as any M is trivially f-DP for $f = 0$ (since the type-II error in hypothesis testing is always ≥ 0).

2.3 Kernel Density Estimation

Kernel density estimation (KDE) is a well-studied tool from non-parametric statistics to approximate an unknown density p by an estimator $\hat{p}$. More concretely, in the presence of sample data $X_1, \ldots, X_n \sim p$ with $X_i \in \mathbb{R}^d$, the KDE for p is given by

$$\hat{p}(t) := \frac{1}{nb^d} \sum_{i=1}^{n} K\Big(\frac{t - X_i}{b}\Big).$$

One can think of the KDE as a smoothed histogram where the bandwidth parameter $b > 0$ corresponds to the bin size for histograms. The kernel function K indicates the weight we assign each observation X_i and is oftentimes taken to be the Gaussian kernel with

$$K(t) = \frac{1}{(2\pi)^{d/2}} \exp\left(-\frac{|t|^2}{2}\right).$$

The appropriate choice of b and K can ensure the uniform convergence of $\hat{p}$ to the true, underlying density p (as in Assumption 2). Higher smoothness of the density p is generally associated with faster convergence rates and we refer to [26] and [41] for a rigorous definition of KDE and associated convergence results.

2.4 Machine Learning Classifiers

Binary classifiers are the final addition to our technical toolbox. We begin with some notations: We denote a generic classifier on the Euclidean space $\mathbb{R}^d$ by ϕ. Formally, a *classifier* is not that different from a statistical test: It is a (potentially random) binary function $\phi : \mathbb{R}^d \to \{0,1\}$. However, its interpretation is different from hypothesis testing, because we do not have a default belief in a label 0 or 1. Let us now consider a probability distribution $\mathcal{P}$ on the combined space of inputs and outputs $\mathbb{R}^d \times \{0,1\}$. A classification error has occurred for a pair $(x,y) \in \mathbb{R}^d \times \{0,1\}$, whenever $\phi(x) \neq y$. If (x,y) are randomly drawn from $\mathcal{P}$, we define the risk of the classifier ϕ w.r.t. to $\mathcal{P}$ as

$$R(\phi) = \Pr_{(x,y)\sim\mathcal{P}}[\phi(x) \neq y].$$

Bayes Classification Problem. The Bayes classification problem refers to a setup to generate the distribution $\mathcal{P}$, where a Bernoulli random variable $Y \in \{0,1\}$ is drawn and then a second variable X with

$$(X|Y=0) \sim P, \qquad (X|Y=1) \sim Q.$$

In our work, we specifically consider the case where Y is drawn from a fair coin flip (i.e., $\Pr[Y=0] = \Pr[Y=1] = \frac{1}{2}$), and we denote this setup by $\mathbf{P}[P,Q]$.

Bayes (Optimal) classifiers. ϕ^* minimizes the risk in the Bayes classification problem. However, ϕ^* is usually unknown in practice because it depends on the (unknown) P and Q. To approximate ϕ^*, one can use a feasible nearest-neighbor classifier [2]. Specifically, a k-nearest neighbors

(k-NN) classifier, denoted as $\phi^{\mathtt{NN}}_{k,n}$, assigns a label to an observation $o \in O$ by identifying its k closest neighbors[3] from the size n training set. The label is then determined by a majority vote among these k neighbors.

The following convergence result for k-NN gauges how close the true risk $R(\phi^{\mathtt{NN}}_{k,n})$ of the k-NN classifier $\phi^{\mathtt{NN}}_{k,n}$ is to the risk of the optimal classifier, $R(\phi^*)$.

Theorem 2.2 (Convergence of k-NN Classifier [17]) *Let $\mathcal{P}$ be a joint distribution with support $O \times \mathcal{Y}$. If the conditional distribution $\mathcal{P}|\mathcal{Y}$ has a density, $O \subseteq \mathbb{R}^d$, and $k = \sqrt{n}$, then for every $\varepsilon > 0$ there is an n_0 such that for $n > n_0$,*

$$\Pr[|R(\phi^{\mathtt{NN}}_{k,n}) - R(\phi^*)| > \varepsilon] \leq 2e^{-n\varepsilon^2/(72c_d^2)},$$

where c_d[4] is the minimal number of cones centered at the origin of angle $\pi/6$ that cover $\mathbb{R}^d$. Note that if the number of dimensions d is constant, then c_d is also a constant.

3 Overview of Techniques

Our goal is to provide an estimation and auditing procedure for the optimal privacy curve f of a mechanism M. This task can be broken down into two parts: (1) Selecting datasets D, D' that cause the largest difference in M's output distributions and (2) Developing an estimator/auditor for the trade-off curve given that choice of D, D'. In line with previous works on black-box estimation/auditing, we focus on task (2). The selection of D, D' has been studied in the black-box setting and can typically be guided by simple heuristics [13, 18, 33].

Our proposed estimator of a trade-off curve relies on KDEs. Density estimation in general and KDE in particular is an important tool in the black box assessment of DP. For some examples, we refer to [32], [6] and [31]. The reason is that DP can typically be expressed as some transformation of the density ratio p/q – this is true for standard DP (a supremum), Rényi DP (an integral) and, as we exploit in this paper, f-DP via the Neyman-Pearson test. A feature of our new approach is that we do not simply plug in our estimators in the definition of f-DP, but rather use them to make a novel, approximately optimal test. This test is not only easier to analyze than the standard likelihood ratio (LR) test but also retains similar properties (see the next section for details).

Our second goal (Sec. 5.2) is to audit whether a mechanism M satisfies a claimed trade-off f, given datasets D and D'. At a high level, we address this task by identifying and studying the *most vulnerable point* on the trade-off curve T of M — the point most likely to violate f-DP. We begin by using our f-DP estimator to compute a value η (from the Neyman-Pearson framework in Sec. 2.1), which defines a point $\big(\alpha(\eta), \beta(\eta)\big)$ on the true privacy curve T of the mechanism M. η is chosen such that $\big(\alpha(\eta), \beta(\eta)\big)$ has the largest distance from the claimed trade-off curve f asymptotically, which we prove in Prop. 4.3. Next, by extending a technique proposed in [34], we express $\big(\alpha(\eta), \beta(\eta)\big)$ in terms of the Bayes risk of a carefully constructed Bayesian classification problem, and approximate that Bayes risk using a feasible binary classifier (e.g., k-nearest neighbors). By deploying the k-NN classifier, we obtain a confidence interval that contains our vulnerable point (α, β) with high probability (in the Appendix, we provide a brief explanation for choosing confidence intervals over the credible intervals used in other works [38, 51]). Finally, our auditor decides whether to reject (or fail to reject) the claimed f curve by checking whether the corresponding point (α, β') on f with $f(\alpha) = \beta'$ is contained in this interval or not. Leveraging the convergence properties of k-NN, our auditor provides a provable and tuneable confidence region that depends on sample size. We also note that the connection between Bayes classifiers and f-DP that underpins our auditor may be of independent interest, as it offers a new interpretation of f-DP by framing it in terms of Bayesian classification problems.

4 Goal 1: f-DP Estimation

In this section, we develop a new method for the approximation of the entire optimal trade-off curve. The trade-off curve results from a study of the Neyman-Pearson test, where any type-I error α is associated with the smallest possible type-II error β (see Section 2.1 for details). Understood as a function in α, we denote the type-II error by $T : [0,1] \to [0,1]$ and call it a trade-off curve. We note that any trade-off curve is continuous, non-increasing and convex (see [19]).

4.1 Estimation of the f-DP curve

Our approach is based on the perturbed likelihood ratio (LR) test which mimics the properties of the optimal Neyman-Pearson test, but requires less knowledge about the distributions involved. In the following, we denote by P, Q the output distributions of $M(D), M(D')$ respectively. The corresponding probability densities are denoted by p, q.

The perturbed LR test. The optimal test for the hypotheses pair

$$H_0 : X \sim p \quad \text{vs.} \quad H_1 : X \sim q$$

is the Neyman-Pearson test described in Section 2.1. It is also called a *likelihood ratio* (LR) test, because it rejects H_0 if the density ratio satisfies $q(X)/p(X) > \eta$ for some

[3] In our context, closeness is measured using Euclidean distance

[4] By Lemma 5.5 of [17], c_d satisfies $c_d \leq (1 + 2/\sqrt{2-\sqrt{3}})^d - 1$.

threshold η. If $q(X)/p(X) = \eta$ the test rejects randomly with probability λ. In a black-box scenario, this process is difficult to mimic, even if two good estimators, say $\hat{p}, \hat{q}$ of p, q are available. Even if $\hat{p} \approx p$ and $\hat{q} \approx q$, it will usually be the case that

$$q(x)/p(x) = \eta \quad \text{does not imply} \quad \hat{q}/\hat{p} = \eta$$

(it may hold that $\hat{p}/\hat{q} \approx \eta$, but typically not exact equality). In principle, one could cope with this problem by modifying the condition $\hat{q}/\hat{p} = \eta$ to $\approx \eta$ to mimic the optimal test. Yet, the implementation of this approach turns out to be difficult. In particular, it would involve two tuneable arguments (η, λ), as well as further parameters (to specify "$\approx$"), making approximations costly and unstable. A simpler and more robust approach is to focus on a different test rather than the optimal one - a test that is close to optimal but does not require the knowledge of when q/p is constant. For this purpose, we introduce here the novel *perturbed LR test* (PLRT). We define it as follows: Let $U \in [-1/2, 1/2]$ be uniformly distributed and $h > 0$ a (small) number. Then we make the decision

$$\text{"reject } H_0 \text{ if} \quad q(X)/p(X) > \eta + hU\text{"}. \tag{2}$$

Just as the Neyman-Pearson test, the perturbed LR test is randomized. Instead of flipping a coin when $q/p = \eta$, the threshold η is perturbed with a small, random noise term. Obviously the perturbed LR test does not require knowledge of the level sets $\{q/p = \eta\}$, making it more practical for our purposes. To formulate a theoretical result for this test, we impose two natural assumptions.

Assumption 1

i) The densities p, q are continuous.

ii) There exists only a finite number of values $\eta \geq 0$ where the set $\{q/p = \eta\}$ has positive mass.

The second assumption is met for all density models that the authors are aware of and in particular for all mechanisms commonly used in DP. Let us denote the f-DP curve of the perturbed LR test by T_h. The next Lemma shows that for small values of h the perturbed LR test performs as the optimal LR test.

Lemma 4.1 *Under Assumption 1 it holds that*

$$\lim_{h \downarrow 0} \sup_{\alpha \in [0,1]} |T(\alpha) - T_h(\alpha)| = 0.$$

Approximating T_h. The Lemma shows that to create an estimator of the optimal trade-off curve T, it is sufficient to approximate the curve T_h of the perturbed LR test for some small h. This is an easier task, since we do not need to know the level sets $\{q/p = \eta\}$ for all η. Indeed, suppose we have two estimators $\hat{p}, \hat{q}$. Then we can run a perturbed LR test with them, just as in equation (2). A short theoretical derivation (found in the appendix) then shows that running the perturbed LR test for $\hat{p}, \hat{q}$ and some threshold η, yields the following type-I and type-II errors:

$$\hat{\alpha}_h(\eta) := \int_{x \in [-h/2, h/2]} \frac{1}{h} \int_{\hat{q}/\hat{p} > \eta + x} \hat{p}, \tag{3}$$

$$\hat{\beta}_h(\eta) := 1 - \int_{x \in [-h/2, h/2]} \frac{1}{h} \int_{\hat{q}/\hat{p} > \eta + x} \hat{q}. \tag{4}$$

The entire trade-off-curve for the perturbed LR test with $(\hat{p}, \hat{q})$ is then given by $\hat{T}_h$ with

$$\hat{T}_h(\alpha) = \hat{\beta}_h(\eta) \quad \Leftrightarrow \quad \alpha = \hat{\alpha}_h(\eta). \tag{5}$$

For the curve estimate $\hat{T}_h$ to be close to T_h (and thus T), the involved density estimators need to be adequately precise. We hence impose the following regularity condition on them. In the condition, n is the sample size used to create the estimators.

Assumption 2 *The density estimators $\hat{p}, \hat{q}$ are themselves continuous probability densities that decay to 0 at $\pm\infty$ (see eq. (15) for a precise definition) . For a null-sequence of non-negative numbers $(a_n)_{n \in \mathbb{N}}$ they satisfy*

$$\Pr[\sup_x |\hat{p}(x) - p(x)| > a_n] = o(1)$$

$$\textit{and} \quad \Pr[\sup_x |\hat{q}(x) - q(x)| > a_n] = o(1).$$

The above assumption is in particular satisfied by KDE (see Section 2.3), where the convergence speed a_n depends on the smoothness of the underlying densities. However, in principle other estimation techniques than KDE could be used, as long as they produce continuous estimators. The next result formally proves the consistency of $\hat{T}_h$. The notation of "$o_P(1)$" refers to a sequence of random variables converging to 0 in probability.

Theorem 4.2 *Suppose that Assumptions 1 and 2 hold, and that $h = h_n$ is a positive number depending on n with $h_n \to 0$ and $h_n/a_n \to \infty$. Then, as $n \to \infty$ it follows that*

$$\sup_{\alpha \in [0,1]} |\hat{T}_h(\alpha) - T(\alpha)| = o_P(1).$$

The above result proves that simultaneously for all α, the curve $\hat{T}_h$ approximates the optimal trade-off function T. Thus, we have achieved the first goal of this work. The (very favorable) empirical properties of $\hat{T}_h$ will be studied in Section 6. We have also incorporated Algorithm 3 for an overview of the procedure in the appendix.

4.2 Finding maximum vulnerabilities

We conclude this section by some preparations for the second goal - auditing f-DP. The precise problem of auditing is described in Section 5.2. Here, we only mention that the task of auditing is to check (in some sense) whether f-DP holds for a claimed trade-off curve, say $f = T^{(0)}$. As an initial step, to check $T^{(0)}$-DP, we create the estimator $\hat{T}_h$ for the optimal curve T. If $T^{(0)}$-DP holds, this means that

$$T(\alpha) \geq T^{(0)}(\alpha) \quad \forall \alpha \in [0,1]. \tag{6}$$

A priori, we cannot say whether this is true or not. However, by comparing our estimator $\hat{T}_h$ with $T^{(0)}$ we can gather some evidence. For example, if $\hat{T}_h(\alpha)$ is much smaller than $T^{(0)}(\alpha)$ for some α, it then seems that the claim in (6) is probably false. We will develop a rigorous criterion for what "much smaller" means in the next section. For now, we will confine ourselves to identifying a point where privacy seems most likely to be broken. We therefore define

$$\hat{\eta}^* \in \operatorname{argmax}\left\{T^{(0)}(\hat{\alpha}_h(\eta)) - \hat{T}_h(\hat{\alpha}_h(\eta)) : \eta \geq 0\right\} \tag{7}$$

and the next result shows that the discrepancy between $T^{(0)}$ and T is indeed maximized in $\hat{\eta}^*$ for large n.

Proposition 4.3 *Suppose that the assumptions of Theorem 4.2 hold. Then, it follows that*

$$\begin{aligned} &T^{(0)}(\hat{\alpha}_h(\hat{\eta}^*)) - T(\hat{\alpha}_h(\hat{\eta}^*)) \\ &= \sup_{\alpha \in [0,1]} \left[T^{(0)}(\alpha) - T(\alpha)\right] + o_P(1). \end{aligned}$$

The threshold $\hat{\eta}^*$ demarcates the greatest weakness of the $T^{(0)}$-privacy claim and it is therefore ideally suited as a starting point for our auditing approach in Section 5.2.

5 Goal 2: Auditing f-DP

In this section, we develop methods for uncertainty quantification in our assessment of T. We begin with Section 5.1, where we derive (two dimensional) confidence regions for a pair of type-I and type-II errors. Our approach relies on the approximation of Bayes optimal classifiers using the k-nearest neighbor (k-NN) method. The resulting confidence regions are used in Section 5.2 as a subroutine of a general-purpose f-DP auditor that combines the estimators from KDE and the confidence regions from k-NN.

5.1 Pointwise confidence regions

In this section, we introduce the BayBox estimator, an algorithm designed to provide point-wise estimates of the trade-off curve T with theoretical guarantees. Specifically, for a given threshold $\eta > 0$, the BayBox estimator outputs an estimate of the trade-off point $(\alpha(\eta), \beta(\eta))$. This estimate is guaranteed to be within a small additive error of the true trade-off point, with high probability.

The BayBox estimator is backed up by the observation that the quantity $\alpha(\eta)$ (also $\beta(\eta)$) can be expressed as the Bayes risk of a carefully constructed Bayesian classification problem. For instance, to compute $\alpha(\eta)$ when $\eta \geq 1$, a theoretical derivation (provided in the appendix) shows that this computation is equivalent to computing the Bayes risk for the Bayesian classification problem $\mathbf{P}\left[[P]_\eta, Q\right]$[5]. The mixture distribution $[P]_\eta$ is formally defined in the following.

Definition 4 (Mixture Distribution) *Let P be a distribution and $\eta \in [1, +\infty)$. The mixture distribution $[P]_\eta$ is defined as:*

$$[P]_\eta = \begin{cases} P & \textit{with probability } \frac{1}{\eta}, \\ \perp & \textit{with probability } 1 - \frac{1}{\eta}. \end{cases}$$

We note that recent work [34] showed that the parameters of approximate DP can be expressed in terms of the Bayes risk of carefully constructed Bayesian classification problems. They further showed how to construct such classification problems using mixture distributions. Building on this foundation, our results significantly extend their approach by establishing a direct link between the theory of optimal classification and f-DP.

Algorithm 1 BayBox: A Black-Box Bayesian Classification Algorithm for f-DP Estimation

Require: Black-box access to M; Threshold $\eta > 0$; Sample size n.

Ensure: An estimate $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ of $(\alpha(\eta), \beta(\eta))$ for tuple (P, Q), where $M(D)$ and $M(D')$ are distributed according to P, Q, respectively.

```
Set the classifier $\phi$ for the Bayesian classification problem $\mathbf{P}\left[[P]_\eta, Q\right]$ if $\eta \geq 1$; otherwise, set $\phi$ for the problem $\mathbf{P}\left[P, [Q]_{1/\eta}\right]$. By default, use the $k$-NN classifier $\phi^{\mathrm{NN}}_{k,n}$ with $k = \sqrt{n}$.
function BayBox Estimator $\mathrm{BB}^{\phi}(M, D, D', \eta, n)$
    Set $cnt_\alpha \leftarrow 0$ and $cnt_\beta \leftarrow 0$
    for $i \in [n]$ do
        $x \leftarrow M(D)$; $x' \leftarrow M(D')$
        If $\phi(x) = 1$ then $cnt_\alpha \leftarrow cnt_\alpha + 1$
        If $\phi(x') = 1$ then $cnt_\beta \leftarrow cnt_\beta + 1$
    end for
    Return $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta)) \leftarrow (\frac{cnt_\alpha}{n}, 1 - \frac{cnt_\beta}{n})$
end function
```

The key insight to connect classification and f-DP is that the trade-off point $(\alpha(\eta), \beta(\eta))$ can be expressed as the expected classification error of the Bayes optimal classifier. We propose a simple Monte Carlo estimator for the expected classification error and an implementation is given by the BayBox estimator in Algorithm 1. In theory,

[5] Refer to Section 2.4 for the notation and setup of the Bayesian classification problem.

if the Bayes optimal classifier ϕ^* were known and used as input of the BayBox algorithm, the output of BayBox would be an unbiased estimator for $(\alpha(\eta), \beta(\eta))$ that has a small error with high probability. A formal statement is provided in Lemma B.2 of the appendix. In practice ϕ^* is unknown, but can be approximated using a k-NN classifier. A statement of the theoretical approximation properties is given in the next theorem. The notation "$\mathbb{E}$" refers to the expectation of a random variable, conditional on the threshold η.

Theorem 5.1 *Suppose that Assumption 1 holds. Let η, $(\alpha(\eta), \beta(\eta))$, $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$, and ϕ be defined as in Algorithm 1. Set ϕ to the k-NN classifier $\phi^{\mathrm{NN}}_{k,n}$, with $k = \sqrt{n}$, for the corresponding Bayesian classification problem :*
1) *Then, for any $\gamma \in (0,1)$ and any $n \geq 2$ it holds with probability $\geq 1 - \gamma$ that*

$$\max\left\{|\tilde{\alpha}(\eta) - \mathbb{E}[\tilde{\alpha}(\eta)]|, |\tilde{\beta}(\eta) - \mathbb{E}[\tilde{\beta}(\eta)]|\right\} \leq w(\gamma),$$

2) *Moreover, for any $\gamma \in (0,1)$ and for all n sufficiently large it holds with probability $\geq 1 - \gamma$ that*

$$\max\left\{|\tilde{\alpha}(\eta) - \alpha(\eta)|, |\tilde{\beta}(\eta) - \beta(\eta)|\right\} \leq (25c_d)w(\gamma) \ ,$$

Here, c_d is a constant depending on the dimension d with $c_d \leq 4.9^d$ and the bound $w(\gamma)$ is defined as

$$w(\gamma) := \sqrt{\ln(4/\gamma)/(2n)} \ . \tag{8}$$

The two statements in the above theorem are distinct and have different interpretations. Part 1) shows that the output of the BayBox algorithm $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ is randomly fluctuating in a narrow region of width $w(\gamma)$ around its expectation $(\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta))$. The expectation $(\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta))$ can be shown to always lie on or above the optimal trade-off curve (Remark 3, in the appendix) and in this sense the output $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ can be slightly biased (it may overstate privacy). Part 2) of the theorem entails the stronger statement that $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ is actually close to the true value $(\alpha(\eta), \beta(\eta))$. The price is a looser bound by a factor of $(25c_d)$, which arises from bounding the distances $|\mathbb{E}\tilde{\alpha}(\eta) - \alpha(\eta)|$ and $|\mathbb{E}\tilde{\beta}(\eta) - \beta(\eta)|$. In principle, auditing mechanisms for f-DP can be based on either part 1) or part 2) of the theorem, and in the next section we give details. Practically, using part 1) yields better results (more accurate detection for lower sample sizes) and will be used in our below methodology. Notice that it is also a finite sample bound and non-asymptotic.

5.2 Auditing f-DP

Outline In the remainder of this section, we present an f-DP auditor that fuses the localization of maximum vulnerabilities (by the KDE method) with the confidence guarantees (afforded by the k-NN method). We can describe the problem as follows: Usually, when a DP mechanism M is developed it comes with a privacy guarantee for users. In the case of standard DP this takes the form of a single parameter ε_0. In the case of f-DP a privacy guarantee is associated with a continuous trade-off curve $T^{(0)}$. Essentially the developer promises that the mechanism will afford at least $T^{(0)}$-DP. The task of the auditor is to empirically and reliably check this claim.

The auditor We proceed in two steps. Since we do not want to force the two steps to depend on the same sample size parameters, we introduce two (potentially different) sample sizes n_1, n_2. First, using the KDE method, we find an estimated value of maximum vulnerability $\hat{\eta}^*$ (based on a sample of size n_1). This is possible according to Proposition 4.3. Second, we apply the BayBox algorithm with input $\hat{\eta}^*$ and sample size n_2, giving us outputs $(\tilde{\alpha}(\hat{\eta}^*), \tilde{\beta}(\hat{\eta}^*))$. Then, we draw on Theorem 5.1 to check the $T^{(0)}$-DP claim. More precisely, recall that the pair $(\mathbb{E}\tilde{\alpha}(\hat{\eta}^*), \mathbb{E}\tilde{\beta}(\hat{\eta}^*))$ lies on or above the optimal, unknown trade-off curve T. This is intuitively clear, because any classifier will have a worse (at best equal) performance as the Bayes optimal classifier, which lies exactly on the curve T. Now, from Theorem 5.1 part 1) we know that the pair $(\mathbb{E}\tilde{\alpha}(\hat{\eta}^*), \mathbb{E}\tilde{\beta}(\hat{\eta}^*))$ is included with high probability inside the box

$$\begin{aligned} \square_\gamma := &\left[\tilde{\alpha}(\hat{\eta}^*) - w(\gamma), \tilde{\alpha}(\hat{\eta}^*) + w(\gamma)\right] \\ &\times \left[\tilde{\beta}(\hat{\eta}^*) - w(\gamma), \tilde{\beta}(\hat{\eta}^*) + w(\gamma)\right]. \end{aligned} \tag{9}$$

This means that $\square_\gamma$ includes points that are above the optimal trade-off curve T with high probability. Now there are two cases: First, if the claim of $T^{(0)}$-DP is true (i.e. $T^{(0)} \leq T$), then some points in $\square_\gamma$ must also be above $T^{(0)}$. For our auditor this means that if there are points in $\square_\gamma$ that are above $T^{(0)}$, we will detect "no privacy violation" (see for an illustration Figure 4). In other words, our findings are compatible with $T^{(0)}$-DP. Conversely, if we observe that the entire box $\square_\gamma$ is below $T^{(0)}$, then our auditor will detect a "privacy violation" and our findings are at odds with $T^{(0)}$-DP (see for an illustration Figure 5). Algorithm 2 summarizes the procedure we have just described. It uses a small geometrical argument to check more easily whether the entire box is below $T^{(0)}$ or not (see lines $6-7$ of the algorithm).

Theoretical analysis To provide theoretical guarantees for the algorithm, we add a mathematical assumption on the trade-off curve of $p \sim M(D), q \sim M(D')$.

Assumption 3 *The optimal trade-off curve T corresponding to the output densities p, q is strictly convex.*

We can now formulate the main theoretical result for the auditor.

Algorithm 2 Privacy Violation Detection Algorithm

```
Require: Mechanism $M$, neighboring databases $D, D'$, sample sizes $n_1, n_2$, confidence level $\gamma$, threshold vector $\eta$, claimed curve $T^{(0)}$.
Ensure: "Violation" or "No Violation".
function Auditor$(M, D, D', n_1, n_2, \gamma, \eta, T^{(0)})$
    Compute $\hat{T}_h$ using $\mathtt{PLRT}^h_{\mathcal{A}}(M, D, D', \eta, n_1)$ for all $\eta_i \in \eta$.
    Compute $\hat{\eta}^* \in \arg\max \left\{ T^{(0)}(\hat{\alpha}_h(\eta)) - \hat{T}_h(\hat{\alpha}_h(\eta)) : \eta \geq 0 \right\}$.
    Run the $k$-NN BayBox estimator $\mathtt{BB}^{\phi^{\mathtt{NN}}_{k,n_2}}(M, D, D', \hat{\eta}^*, n_2)$ to obtain $(\tilde{\alpha}(\hat{\eta}^*), \tilde{\beta}(\hat{\eta}^*))$.
    Calculate the threshold $w(\gamma)$ from eq. (8)
    Calculate $i^*$ as the solution to $T^{(0)}(i^*) = \tilde{\beta}(\hat{\eta}^*) + w(\gamma)$.
    if $i^* > \tilde{\alpha}(\hat{\eta}^*) + w(\gamma)$ then
        return "Violation".
    else
        return "No Violation".
    end if
end function
```

Theorem 5.2 *Suppose that Assumptions 1 and 2 hold, let $\gamma \in (0,1)$ be user-determined and denote the output of Auditor($M, D, D', n_1, n_2, \gamma, \eta$, $T^{(0)}$) by A.*

1) *Then, if $T^{(0)}(\alpha) \leq T(\alpha)$ for all $\alpha \in [0,1]$ (no violation of $T^{(0)}$-DP), it follows for any $n_1, n_2 \geq 2$*

$$\Pr\left[A = \text{"No Violation"}\right] \geq 1 - \gamma.$$

2) *Suppose that additionally Assumption 3 holds. Then, if $T^{(0)}(\alpha^*) > T(\alpha^*)$ for some $\alpha^* \in [0,1]$ (a violation of $T^{(0)}$-DP), it follows that*

$$\lim_{n_1 \to \infty} \liminf_{n_2 \to \infty} \Pr\left[A = \text{"Violation"}\right] = 1.$$

Part 1) of the theorem states that the risk of falsely detecting a violation can be made arbitrarily small ($\leq \gamma$) by the user. On the other hand, if some violation exists, part 2) assures that it will be reliably detected for large enough sample sizes. We note that for smaller values of γ larger sample sizes are typically needed to detect violations. This follows from the definition of the box $\square_\gamma$ in (9).
The theoretical Assumptions 1-2 of the theorem are comparable to related works [31, 34] and require smoothness of the output distributions p, q. Such assumptions are required to avoid known impossibility results in privacy estimation (see [22]). Assumption 3 of a strictly convex trade-off function is often satisfied (e.g. for Gaussian type mechanisms), but can be further relaxed. A simple to prove but fairly general relaxation is that T is only strictly convex in a sufficiently small, open neighborhood of the set *argmax*$(T^{(0)} - T)$. We do not include it here, to avoid making the results even more technical.

Remark 1 *The auditor in Algorithm 2 uses the threshold $\hat{\eta}^*$ (see eq. 7), to locate the maximum vulnerability. We point out that any other method to find vulnerabilities would still enjoy the guarantee from part 1) of Theorem 5.2 (it is a property of k-NN), but not necessarily of part 2). It might be an interesting subject of future work to consider other ways of choosing $\hat{\eta}^*$ (e.g. based on the two dimensional Euclidean distance between $T^{(0)}$ and $\hat{T}_h$ rather than the supremum distance).*

Remark 2 *Our black-box algorithms for estimation and auditing face computational limitations due to sample size requirements and the curse of dimensionality. These challenges arise from the black-box setting itself, where we require larger amounts of data samples to make up for missing information and knowledge with regard to algorithm structure. In addition, higher dimensional algorithm output only increases the need for larger sampling efforts. This makes the auditing of machine learning on large, real-world datasets challenging. Here, white-box methods that aim at minimizing the amount of trained models needed for a privacy audit can be of help (see e.g. [38]). We discuss how modifications of our algorithms could help with these limitations in Section 8.*

6 Experiments

We investigate the empirical performance of our new procedures in various experiments to demonstrate their effectiveness. Recall that our procedures are developed for two distinct goals, namely estimation of the optimal trade-off curve T (see Section 4) and auditing a privacy claim $T^{(0)}$ (see Section 5). We will run experiments for both of these objectives.
Experiment Setting: Throughout the experiments, we consider databases $D, D' \in [0,1]^r$, where the participant number is always $r = 10$. As discussed in Section 3, we first choose a pair of neighboring datasets such that there is a large difference in the output distributions of $M(D)$ and $M(D')$. We can achieve this by simply choosing D and D' to be as far apart as possible (while still remaining neighbors) and we settle on the choice

$$D = (0, \ldots, 0) \quad \text{and} \quad D' = (1, 0, \ldots, 0) \tag{10}$$

for all our experiments.

6.1 Mechanisms

In this section, we test our methods on two frequently encountered mechanisms from the auditing literature: the Gaussian mechanism and differentially private Stochastic Gradient Descent (DP-SGD). We study two other prominent DP algorithms – the Laplace and Subsampling mechanism – in Appendix B.

Gaussian mechanism. We consider the summary statistic $S(x)=\sum_{i=1}^{10} x_i$ and the mechanism

$$M(x) := S(x) + Y ,$$

where $Y \sim \mathcal{N}(0,\sigma^2)$. The statistic $S(x)$ is privatized by the random noise Y if the variance σ^2 of the Normal distribution is appropriately scaled. We choose $\sigma = 1$ for our experiments and note that - in our setting - the optimal trade-off curve is given by

$$T_{Gauss}(\alpha) = \Phi(\Phi^{-1}(1-\alpha) - \mu)$$

with $\mu = 1$. We point the reader to [19] for more details.

DP-SGD. The DP-SGD mechanism is designed to (privately) approximate a solution for the empirical risk minimization problem

$$\theta^* = argmin_{\theta\in\Theta}\mathcal{L}_x(\theta) \quad \text{with} \quad \mathcal{L}_x(\theta) = \frac{1}{r}\sum_{i=1}^{r} \ell(\theta, x_i) \ .$$

Here, ℓ denotes a loss function, Θ a closed convex set and $\theta^* \in \Theta$ the unique optimizer. For sake of brevity, we provide a description of DP-SGD in the appendix (see Algorithm 4). In our setting, we consider the loss function $\ell(\theta,x_i) = \frac{1}{2}(\theta - x_i)^2$, initial model $\theta_0 = 0$ and $\Theta = \mathbb{R}$. The remaining parameters are fixed as $\sigma = 0.2, \rho = 0.2, \tau = 10, m = 5$. In order to have a theoretical benchmark for our subsequent empirical findings, we also derive the theoretical trade-off curve T_{SGD} analytically for our setting and choice of databases (see Appendix B for details). Our calculations yield

$$T_{SGD}(\alpha) = \sum_{I\subset\{1,\ldots,\tau\}} \frac{1}{2^\tau}\Phi\left(\Phi^{-1}(1-\alpha) - \frac{\mu_I}{\bar{\sigma}}\right) ,$$

where μ_I is chosen as in (14) and $\bar{\sigma}$ as in (13).

6.2 Simulations

We begin by outlining the parameter settings of our KDE and k-NN methods for our simulations. We then discuss the metrics employed to validate our theoretical findings and, in a last step, present and analyze our simulation results.

Parameter settings: For the KDEs, we consider different sample sizes of $n_1 = 10^2, 10^3, 10^4, 10^5, 10^6$ and we fix the perturbation parameter at $h = 0.1$. For the bandwidth parameter b (see Sec. 2.3), we use the method of [42]. To approximate the optimal trade-off curve, we use 1000 equidistant values for η between 0 and 15 (see Algorithm 3 for details on the procedure). For the k-NN, we set the training sample size to $n_2 = 10^3, 10^4, 10^5$ and testing sample size to $10^3, 10^4$ and 10^5.

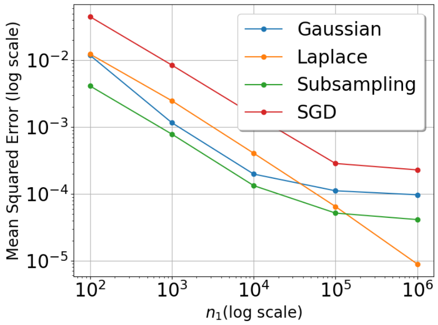


Figure 1: MSE defined in (11) to empirically validate Theorem 4.2 for varying sample sizes n_1 and over 1000 simulation runs each.

Estimation The first goal of this work is estimation of the optimal trade-off curve T. In our experiments, we want to illustrate the uniform convergence of the estimator $\hat{T}_h$ to the optimal curve T, derived in Theorem 4.2. Therefore, we consider increasing sample sizes n_1 to study the decreasing error. The distance of $\hat{T}_h$ and T in each simulation run is measured by the uniform distance[6]

$$Error_T := \sup_{\alpha\in[0,1]} |\hat{T}_h(\alpha) - T(\alpha)|.$$

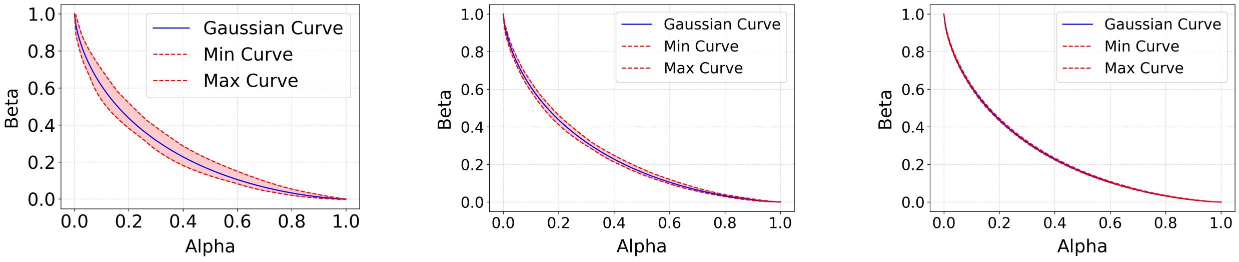
To study not only the distance in one simulation run, but across many, we calculate $Error_T$ in 1000 independent runs and take the (empirical) mean squared error

$$MSE(Error_T) := \mathbb{E}\left[Error_T^2\right] . \qquad (11)$$

The results are depicted in Figure 1 for the DP algorithms described in this section and the appendix. On top of that, we also construct figures that upper and lower bound the worst case errors for the Gaussian mechanism and DP-SGD over the 1000 simulation runs. These plots visually show how the error of the estimator $\hat{T}_h$ shrinks as n_1 grows. The results are summarized in Figures 2-3.

Inference Next, we turn to the second goal of this work: Auditing a $T^{(0)}$-DP claim for a postulated trade-off curve $T^{(0)}$. The theoretical foundations of our auditor can be found in Theorem 5.2. The theorem makes two guarantees: First, that for a mechanism M satisfying $T^{(0)}$-DP the auditor will (correctly) not detect a violation, except with low, user-determined probability γ. Second, if M violates $T^{(0)}$-DP, the auditor will (correctly) detect the violation for sufficiently large sample sizes n_1, n_2. Together, these results mean that if a violation of $T^{(0)}$-DP is detected by the auditor, the user can have high confidence that M

[6] Of course, one cannot practically maximize over all (infinitely many) arguments $\alpha \in [0,1]$. The estimator $\hat{T}_h$ is made for a grid of values for η (see our parameter settings above) and we maximize over all gridpoints.

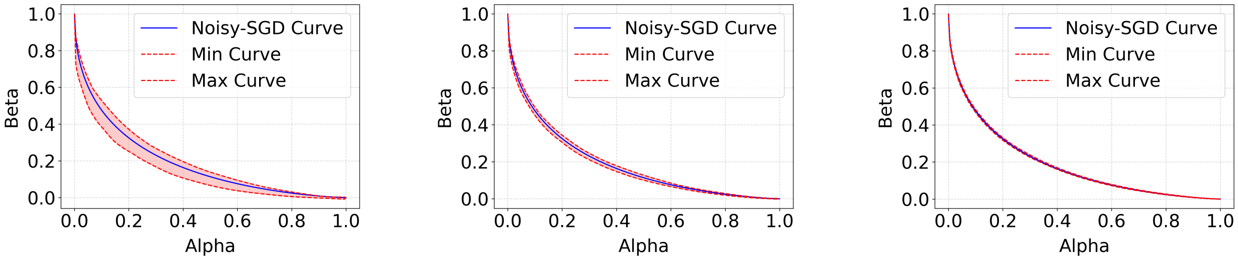


(a) $n_1 = 10^3$

(b) $n_1 = 10^4$

(c) $n_1 = 10^5$

Figure 2: Estimation of the Gaussian Trade-off curve T_{Gauss} for varying sample sizes and $\mu = 1$. Min- and Max Curve lower- and upper bound the worst point-wise deviation from the true curve T_{Gauss} over 1000 simulations.



(a) $n_1 = 10^3$

(b) $n_1 = 10^4$

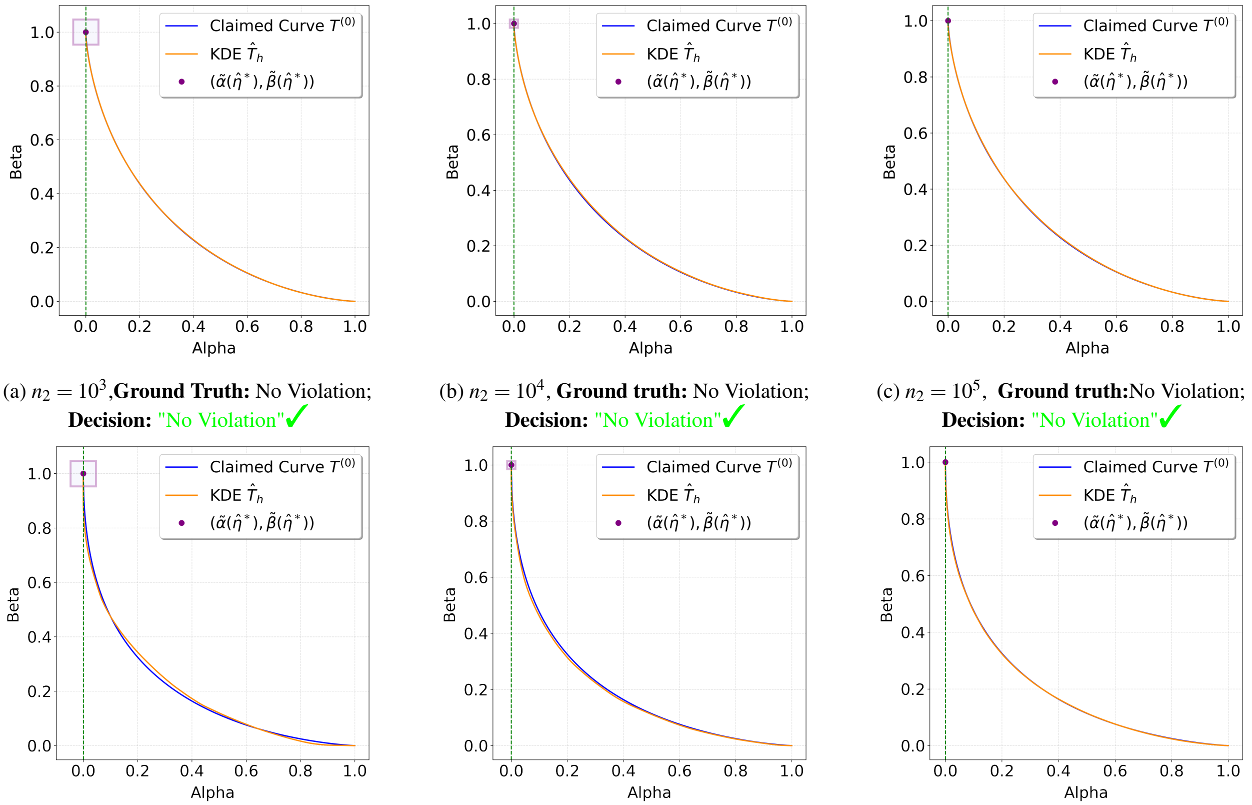
(c) $n_1 = 10^5$

Figure 3: Estimation of the DP-SGD Trade-off curve T_{SGD} for varying sample sizes. Min- and Max Curve lower- and upper bound the worst point-wise deviation from the true curve T_{SGD} over 1000 simulations.

does indeed not satisfy $T^{(0)}$-DP. For the first part, we consider a scenario, where the claimed trade-off curve $T^{(0)}$ is the correct one $T^{(0)} = T$ (M does not violate $T^{(0)}$-DP). For the second part, we choose a function $T^{(0)}$ above the true curve T (M violates $T^{(0)}$-DP). We will consider both scenarios for the Gaussian mechanism and DP-SGD. We run our auditor (Algorithm 2) with parameters $n_1 = 10^4$ and $\gamma = 0.05$ fixed. The choice of $\gamma = 0.05$ is standard for confidence regions in statistics and we further explore the impact of n_1 and γ in additional experiments in Appendix B. Here, we focus on the most impactful parameter, the sample size n_2 and study values of $n_2 = 10^3, 10^4, 10^5$.
Technically, the auditor only outputs a binary response that indicates whether a violation is detected or not. However, in our below experiments, we depict the inner workings of the auditor and geometrically illustrate how a decision is reached. More precisely, in Figure 4 we depict the claimed trade-off curve $T^{(0)}$ as a blue line. The auditor makes an estimate for the true trade-of curve T, namely $\hat{T}_h$ depicted as the orange line. The location, where the orange line (estimated DP) and the blue line (claimed DP) are the furthest apart is indicated by the vertical, dashed green line. This position is associated with the threshold $\hat{\eta}^*$ in Algorithm 2. As a second step, $\hat{\eta}^*$ is used in the kNN method to make a confidence region, depicted as a purple square (this is $\square_\gamma$ from (9)). If the square is fully below the claimed curve $T^{(0)}$, a violation is detected (Figure 5) and if not, then no violation is detected (Figures 2 and 3). As we can see, detecting violations requires n_2 to be large enough, especially when $T^{(0)}$ and T are close to each other.
For the incorrect $T^{(0)}$-DP claims, we have done the following: For the Gaussian case (Figure 5), we have used a trade-off curve with parameter $\mu = 0.5$ instead of the true $\mu = 1$. For DP-SGD, we have used the trade-off curve corresponding to $\tau = 5$ instead of the true $\tau = 10$ iterations (Figure 5).

Implementation Details The implementation is done using python and R. [7]. For the simulations, we have used a local device and a server. All runtimes were collected on a local device with an Intel Core i5-1135G7 processor (2.40 GHz), 16 GB of memory, and running Ubuntu 22.04.5, averaged over 10 simulations. Thus, we demonstrate reasonable runtimes even on a standard personal computer (see Appendix B.4). Additionally, we used a server with four AMD EPYC 7763 64-Core (3.5 GHz) processors and 2 TB of memory running Ubuntu 22.04.4 for repetitive simulations. For python, we have used Python 3.10.12 and the libraries "numpy" [23], "scikit-learn" [40] and "scipy" [46]. For R, we used R version 4.3.1 and the libraries "fdrtool" [28] and "Kernsmooth" [47].

[7] https://github.com/stoneboat/fdp-estimation



(a) $n_2 = 10^3$,**Ground Truth:** No Violation; **Decision:** "No Violation"✓

(b) $n_2 = 10^4$, **Ground truth:** No Violation; **Decision:** "No Violation"✓

(c) $n_2 = 10^5$, **Ground truth:**No Violation; **Decision:** "No Violation"✓

(d) $n_2 = 10^3$, **Ground truth:** No Violation; **Decision:** "No Violation"✓

(e) $n_2 = 10^4$, **Ground truth:** No Violation; **Decision:** "No Violation"✓

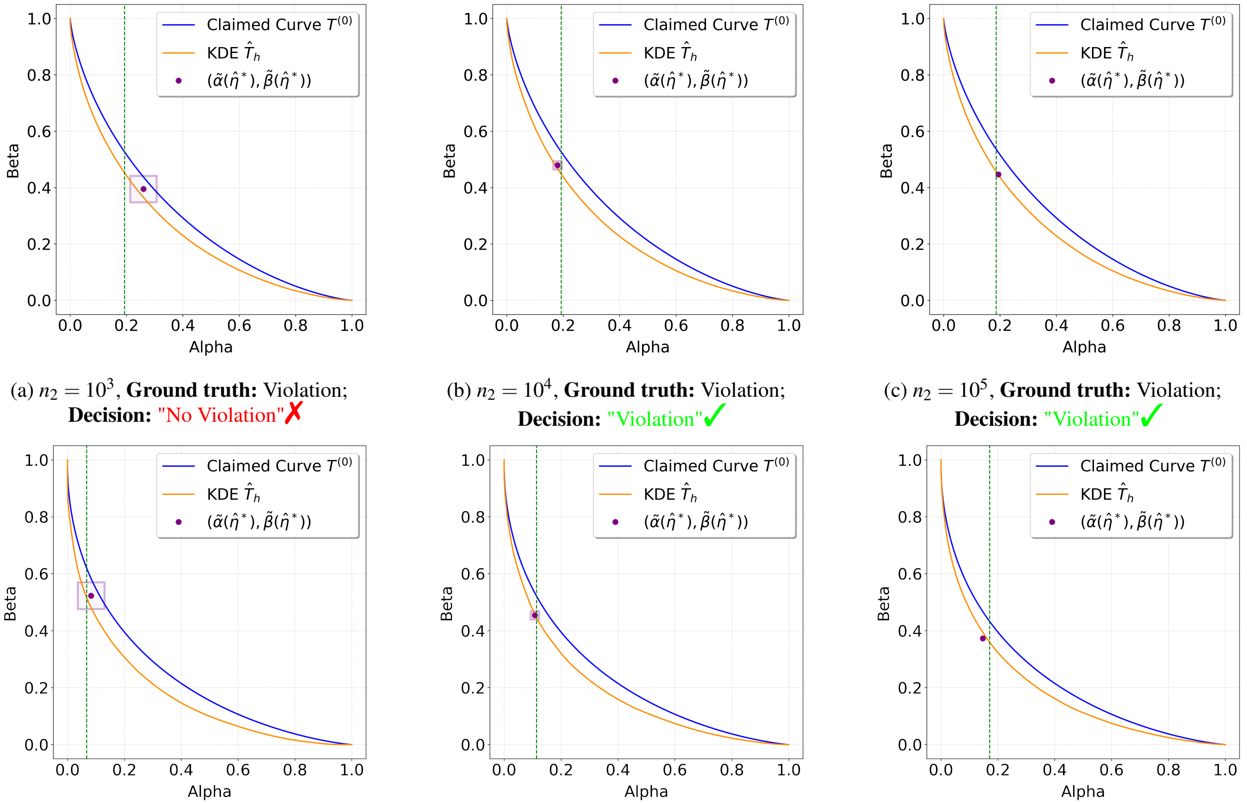
(f) $n_2 = 10^5$, **Ground truth:** No Violation; **Decision:** "No Violation"✓

Figure 4: **Auditing a correct Mechanism:** Claimed curve $T^{(0)} = T_{Gauss}$ (a,b,c) and $T^{(0)} = T_{SGD}$ (d,e,f). We depict the critical vertical line (obtained with step 3 in Algorithm 2) with intercept $(\hat{\alpha}(\hat{\eta}^*), \hat{\beta}(\hat{\eta}^*))$, the k-NN point estimator ● $(\tilde{\alpha}(\hat{\eta}^*), \tilde{\beta}(\hat{\eta}^*))$ and the confidence region □. The sample size for the KDE is $n_1 = 10^4$ and the confidence parameter is $\gamma = 0.05$.

6.3 Interpretation of the results

For Goal 1 (estimation), we see in Figure 1 the fast decay of the estimation error of $\hat{T}_h$ for the optimal trade-off curve. The estimation error decays quickly in n_1, regardless of whether there are plateau values in the sense of Assumption 1 (e.g. Laplace Mechanism) or not (e.g. Gaussian Mechanism). These quantitative results are supplemented by the visualizations in Figures 2–3, where we depict the largest distance of $\hat{T}_h$ and T in 1000 simulation runs (captured by the red band). Even for the modest sample size of $n_1 = 10^3$, this band is fairly tight and for $n_1 = 10^5$ the estimation error is almost too minute to plot. We find this convergence astonishingly fast. It may be partly explained by the estimator $\hat{T}_h$ being structurally similar to T - after all $\hat{T}_h$ is also designed to be a trade-off curve for an almost optimal LR test. The approximation over the entire unit interval corresponds to the uniform convergence guarantee in Theorem 4.2.

For Goal 2 (inference), we recall that a $T^{(0)}$-DP violation is detected if the box $\square_\gamma$ (purple) lies completely below the postulated curve $T^{(0)}$ (blue). In Figure 4, we consider the case of no violation where $T = T^{(0)}$, and we expect not to detect a violation. This is indeed what happens, since $\square_\gamma$ intersects with the curve $T^{(0)}$ in all considered cases. Interestingly, we observe that $\square_\gamma$ has a center close to $\alpha = 0$ in the cases where no violation occurs (such a behavior might give additional visual evidence to users that no violation occurs). In Figure 5, we display the case of faulty claims, where the privacy breach is caused by a smaller variance for both mechanisms under investigation. In accordance with Theorem 5.2, we expect a detection of a violation if n_2 is large enough. This is indeed what happens, at a sample size of $n_2 = 10^4$ for the Gaussian mechanism and at $n_2 = 10^4$ for DP-SGD. Note that larger samples n_2 are needed to expose claims $T^{(0)}$ that are closer to the truth T (as for DP-SGD in our example). For larger n_2 the square $\square_\gamma$



(a) $n_2 = 10^3$, **Ground truth:** Violation; **Decision:** "No Violation"✗

(b) $n_2 = 10^4$, **Ground truth:** Violation; **Decision:** "Violation"✓

(c) $n_2 = 10^5$, **Ground truth:** Violation; **Decision:** "Violation"✓

(d) $n_2 = 10^3$, **Ground truth:** Violation; **Decision:** "No Violation"✗

(e) $n_2 = 10^4$, **Ground truth:** Violation; **Decision:** "Violation"✓

(f) $n_2 = 10^5$, **Ground truth:** Violation; **Decision:** "Violation"✓

Figure 5: **Auditing a faulty Mechanism:** Claimed Curve $T^{(0)} = T_{Gauss}$ (a,b,c) with $\mu = 0.5$ and $T^{(0)} = T_{SGD}$ (d,e,f) with $\tau = 5$. Both mechanisms assume stronger privacy ($\mu = 0.5 < 1$ and $\tau = 5 < 10$). We depict the critical vertical line (obtained with step 3 in Algorithm 2) with intercept $(\hat{\alpha}(\hat{\eta}^*), \hat{\beta}(\hat{\eta}^*))$, the k-NN point estimator ● $(\tilde{\alpha}(\hat{\eta}^*), \tilde{\beta}(\hat{\eta}^*))$ and the confidence region □. The sample size for KDE is $n_1 = 10^4$ and the confidence parameter is $\gamma = 0.05$.

shrinks (see eq. (9)), leading to a higher resolution of the auditor.

6.4 Real-World Example - CIFAR-10

We consider the CIFAR-10 dataset and train a small private convolutional neural network using Opacus[8] [50], a standard library for training PyTorch models with DP.

Parameter settings: To simulate a pair of neighboring databases, we take the same subset of size 1,000 of the CIFAR-10 training data and replace the index 0 by differing images. In D, we use an all-black (all-zero) synthetic image, while in group D', we use an all-white (all-255) image. Both are labeled arbitrarily as "airplane". We train a 4-layer convolutional neural network on 1,000 images, using a batch size of $m = 512$ (so in total 2 batches) for 1,10,15,20 and 25 epochs. We set the learning rate to $\rho = 0.1$, the clipping parameter to 1.0 and use $\sigma = 1.0$ for the noise multiplier. We train the model 1,000 times on D and D' respectively, which took approximately 7 hours on a machine with 64 CPU cores while training in parallel.

Score: For classical neural network structures, taking a reasonable score function will be essential for proper auditing. In image classification, assigning the correct label to the distinct image can yield reasonable results. In our black-box setting, we tried different scores, namely the logits, the cross-entropy (CE), with which the model is trained, and the Kullback Leibler divergence (KL). It is also crucial for our methodology that the score function is indeed one dimensional, as the KDEs performance heavily degrades for higher dimensions. In Figure 6 and Figure 7, we depict the performance of estimation and auditing after 25 epochs. For auditing we have used the KL loss. In Figure 8, we illustrate how tight the lower bounds are over various epochs, which yield different

[8] https://github.com/pytorch/opacus

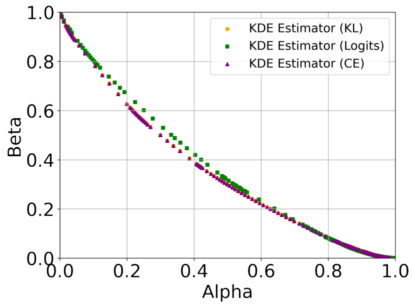


Figure 6: Algorithm 3 based on 1,000 models for different loss functions. KL and CE are overlapping.

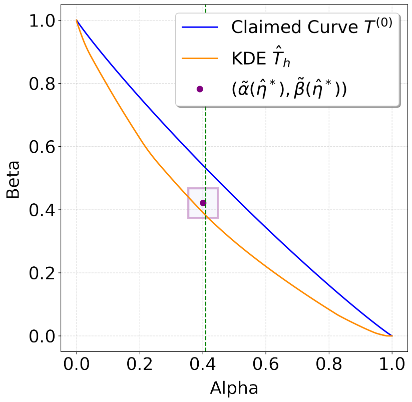


Figure 7: Algorithm 2 based on 1,000 models using the KL. $T^{(0)}$ is a gaussian trade-off curve with $\mu = 0.15$. **Decision:** "Violation".

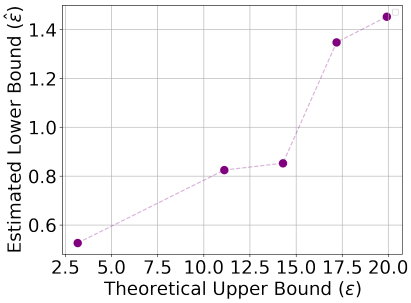


Figure 8: Comparison of theoretical ε and empirical lower bound for $\delta = 0.001$.

theoretical ε.

Interpretation of results: In Figure 6, we compare several scoring functions for estimation. As expected, CE performs best, since the model is trained to minimize it. Nonetheless, related losses (e.g., KL) perform similarly, underscoring the black-box nature of our method. In principle, one could try multiple scores and select the most promising. For auditing (see Figure 7), we consider a misspecified training process, which can result from incorrect noise calibration, excessive training epochs, or misconfigured clipping. We also compare the empirical lower bound $\hat{\varepsilon}$ and theoretical ε obtained using Opacus. We compute the empirical lower bound from the estimated trade-off curve. Specifically, we use binary search to find the Gaussian trade-off function that is closest to our estimated curve in terms of ℓ_1 distance. We then leverage the known relationship between approximate differential privacy and the Gaussian trade-off function to convert this into a lower bound on ε. Similar to [38], we see that the estimates are loose (see Figure 8). Consequently, using the proposed method for auditing is only possible for relatively strong privacy violations. More accurate models, on the other hand, allow for tighter estimates an better auditing, as already highlighted in [38].

7 Related Work

In this section, we provide a more detailed overview of related works on auditing privacy guarantees and f-DP. One avenue to assessing f-DP is to resort to a method that provides estimates for the (ε, δ)-parameter of M and to subsequently exploit the link between standard and f-differential privacy to obtain an estimate of f. To be more concrete, an algorithm that is (ε, δ)-DP is also $f_{\varepsilon,\delta}$-DP (see [19]) with trade-off function

$$f_{\varepsilon,\delta}(\alpha) := \max\left\{0, 1-\delta-e^{\varepsilon}\alpha, e^{-\varepsilon}(1-\delta-\alpha)\right\}. \quad (12)$$

Thus, an estimator for (ε, δ) could, in principle, also provide an estimate for the trade-off curve f of M. This approach is pursued in [30] with the help of a black-box estimator of δ for fixed $\varepsilon > 0$. The maximum over the $f_{\varepsilon,\delta}(\cdot)$-estimates then provides an approximation of the trade-off curve of M. In contrast, our auditing procedure is based on a single, direct estimate of the trade-off curve of M, which makes our approach more expedient. In fact, the runtimes reported for estimation of f in Appendix B.4 confirm the efficacy of our approach. Moreover, from an auditing perspective, results with regard to convergence and reliability in [30] are only obtained for the $\delta(\varepsilon)$-estimate in the standard DP framework. Our work, on the other hand, provides formal statistical guarantees for the inference of the trade-off f.

Interestingly, the relation between standard and f-DP can also be exploited in the opposite direction, that is, to use estimates of the trade-off curve f to obtain estimates for (ε, δ). This approach was first taken in [38] and subsequently adopted in other works [3, 4, 36] for the purpose of auditing the privacy claims of DP-SGD, a cornerstone of differentially private machine learning. Essentially, these works aim at auditing DP by converting confidence intervals for the type-I and type-II error of a distinguishing attack into estimates of (ε, δ). And while previous works [25, 43, 51] investigated the type-I and type-II errors of distinguishing attacks in the standard DP model to obtain such estimates, [38] was the first to exploit the tight characterization of these errors in the Gaussian DP model to obtain even tighter lower bounds for ε. Auditing in [38] focuses on the white-box scenario, where the auditor does not only have access to the training datasets D and D', but can also examine all intermediate model updates that go into computing the corresponding final models θ and θ'. This setting is further enhanced by allowing the auditor to actively intervene in the training process via self-crafted gradients or datasets that can be inserted into the computations that yield the final model outputs. "Opening the black-box" in this manner results in tighter empirical estimates with fewer observations. A black-box scenario discussed in [38] restricts the available information about the instance of DP-SGD under investigation. In this setting, the auditor's access is limited to the training datasets, the corresponding final models and the specific loss function ℓ that the training algorithm uses. Important parameters and features such as model weights, noise scales, sampling rates, learning rates and etc. remain hidden from the auditor. This scenario can thus be considered "parameter black-box". Though far more restrictive than the above white-box setting, this scenario is also specific to the DP-SGD mechanism, which is characterized by the repeated use of Gaussian noise. It therefore allows for auditing procedures that work with the class of (subsampled) Gaussian trade-off curves as in [38] or for distinguishing attacks such as LiRA [15], which models the distribution of losses as Gaussians. Subsequent works have adopted the approach in [38] to study differentially private synthetic data generation [5], the impact of shuffling on the privacy of DP-SGD [3], how to obtain tighter audits in the black-box setting of [38] with specially crafted worst-case parameters [4], or to improve the analysis of distinguishing attacks [36]. In our work, we further tighten the parameter black-box scenario in [38] by avoiding assumptions that a specific mechanism or family of trade-off curves are under investigation. Our approach can thus be deemed "mechanism black-box" and is more aligned with a number of existing works that study other common variants of DP through this lens [6, 13, 22, 29, 31, 34]. With no knowledge of the underlying privacy mechanism, we cannot assume normally distributed output data in this setting. Hence, we developed tools like the perturbed LR test and BayBox estimator, which do not require modeling distributions as Gaussians and instead rely on mild regularity assumptions (like smoothness) that are common in the mechanism black-box setting [6, 22, 34]. Even though it was designed for the more restrictive mechanism black-box scenario, our method's performance is similar to that of the black-box approach from [38] in our experiments on more realistic datasets. Moreover, our approach compares favorably to other works that operate in the same mechanism black-box setting. Here, the required number of output samples to achieve reasonable levels of accuracy usually surpasses the maximum sample size $n = 10^5$ in our experiments on auditing [6, 22, 29] and can even reach into the millions [13, 31, 34]. Thus, our estimation and auditing methods are effective and flexible tools that add to the existing literature on DP validation.

8 Conclusion and Future Work

In our work, we construct the first general-purpose f-DP estimator and auditor for the mechanism black-box setting, by combining techniques from statistics and classification theory. Our constructions enjoy not only formal guarantees—convergence of our estimator and a tuneable confidence region for our auditor—but also perform well in experiments on standard algorithms from the DP literature. Our approach has limitations as well. In practice, training even a single machine learning model can be costly, making it impractical to sample thousands. This is a problem in black-box settings, which require larger samples sizes to achieve desired levels of accuracy.

However, our approach benefits from a plug-and-play design, allowing these limitations to be mitigated by substituting alternative estimators – such as neural network-based estimators – that might be more effective in high-dimensional settings. Replacing the k-NN classifier does not affect the theoretical guarantees provided in Theorem 5.2, although it may increase the chance of failing to reject a false claim if the alternative is not well chosen. For the PLRT component, adopting a density estimator that satisfies Assumption 2 preserves all guarantees established in Theorem 4.2; otherwise, the guarantees may no longer hold. In general, designing improved estimators and classification tools for the black-box scenario is a promising direction for future work.

9 Acknowledgments

This work was funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy - EXC 2092 CASA - 390781972. Tim Kutta was partially funded by the AUFF Nova Grant 47222. Yun Lu is supported by NSERC (RGPIN-03642-2023). Vassilis Zikas and Yu Wei are supported in part by Sunday Group, Incorporated. Holger Dette was partially supported by the DFG Research unit 5381 *Mathematical Statistics in the Information Age*, project number 460867398.

Ethics considerations

We attest that the algorithms our privacy auditor/estimator identify as having privacy vulnerabilities are already recognized for (or specifically created with) these issues. Thus, this work does not expose new vulnerabilities. Additionally, we do not make use of private datasets in our experiments. Thus, our experiment results do not introduce privacy leaks. We attest that our auditor/estimator do not have any biases stemming from a conflict of interest. While our auditor can be used on other existing mechanisms, our implementation mitigates risks by ensuring our output is limited to alerts to potential privacy violation, and does not leak any additional information about the dataset(s) tested.

Open science

To comply with the Open Science Policy, we will make all artifacts publicly available[9]. In our experiments section, we ensure the transparency and reproducibility of our methodology by describing the dataset used, the specification of our machines, and citing all algorithms tested.

References

[1] ABOWD, J. M. The U.S. census bureau adopts differential privacy. In *KDD'18* (2018), ACM, p. 2867.

[2] ALTMAN, N. S. An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician 46*, 3 (1992), 175–185.

[3] ANNAMALAI, M. S. M. S., BALLE, B., CRISTOFARO, E. D., AND HAYES, J. To shuffle or not to shuffle: Auditing DP-SGD with shuffling. *arXiv:2411.10614* (2024).

[4] ANNAMALAI, M. S. M. S., AND CRISTOFARO, E. D. Nearly tight black-box auditing of differentially private machine learning. In *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems, NeurIPS* (2024).

[5] ANNAMALAI, M. S. M. S., GANEV, G., AND CRISTOFARO, E. D. "what do you want from theory alone?" experimenting with tight auditing of differentially private synthetic data generation. In *33rd USENIX Security Symposium* (2024).

[6] ASKIN, Ö., KUTTA, T., AND DETTE, H. Statistical quantification of differential privacy: A local approach. In *SP'22* (2022).

[7] BARTHE, G., FONG, N., GABOARDI, M., GRÉGOIRE, B., HSU, J., AND STRUB, P. Advanced probabilistic couplings for differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS)* (2016).

[8] BARTHE, G., GABOARDI, M., ARIAS, E. J. G., HSU, J., KUNZ, C., AND STRUB, P. Proving differential privacy in hoare logic. In *IEEE 27th Computer Security Foundations Symposium (CSF)* (2014).

[9] BARTHE, G., GABOARDI, M., ARIAS, E. J. G., HSU, J., ROTH, A., AND STRUB, P. Higher-order approximate relational refinement types for mechanism design and differential privacy. In *Proceedings of the 42nd Annual ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)* (2015).

[10] BARTHE, G., GABOARDI, M., GRÉGOIRE, B., HSU, J., AND STRUB, P. Proving differential privacy via probabilistic couplings. In *Proceedings of the 31st Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (2016).

[11] BARTHE, G., KÖPF, B., OLMEDO, F., AND BÉGUELIN, S. Z. Probabilistic relational reasoning for differential privacy. In *Proceedings of the 39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)* (2012).

[12] BICHSEL, B., GEHR, T., DRACHSLER-COHEN, D., TSANKOV, P., AND VECHEV, M. Dp-finder: Finding differential privacy violations by sampling and optimization. In *CCS'18* (2018).

[13] BICHSEL, B., STEFFEN, S., BOGUNOVIC, I., AND VECHEV, M. T. Dp-sniper: Black-box discovery of differential privacy violations using classifiers. In *SP'21* (2021).

[14] BICKEL, P., AND DOKSUM, K. *Mathematical Statistics: Basic Ideas and Selected Topics*. Prentice Hall, 2001.

[15] CARLINI, N., CHIEN, S., NASR, M., SONG, S., TERZIS, A., AND TRAMÈR, F. Membership inference attacks from first principles. In *43rd IEEE Symposium on Security and Privacy, (SP)* (2022).

[16] CHADHA, R., SISTLA, A. P., VISWANATHAN, M., AND BHUSAL, B. Deciding differential privacy of online algorithms with multiple variables. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS)* (2023).

[17] DEVROYE, L., GYÖRFI, L., AND LUGOSI, G. *A Probabilistic Theory of Pattern Recognition*, vol. 31 of *Stochastic Modelling and Applied Probability*. Springer, 1996.

[18] DING, Z., WANG, Y., WANG, G., ZHANG, D., AND KIFER, D. Detecting violations of differential privacy. In *CCS'18* (2018).

[19] DONG, J., ROTH, A., AND SU, W. J. Gaussian differential privacy. *Journal of the Royal Statistical Society Series B: Statistical Methodology 84* (2022).

[20] DWORK, C. Differential privacy. In *Automata, Languages and Programming, 33rd International Colloquium (ICALP)* (2006), Lecture Notes in Computer Science, Springer.

[21] ERLINGSSON, Ú., PIHUR, V., AND KOROLOVA, A. RAPPOR: randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security (CCS)* (2014).

[22] GORLA, D., JALOUZOT, L., GRANESE, F., PALAMIDESSI, C., AND PIANTANIDA, P. On the (im)possibility of estimating various notions of differential privacy. In *Proceedings of the 24th Italian Conference on Theoretical Computer Science (ICTCS 2023)* (Palermo, Italy, 2023), vol. 3587 of *CEUR Workshop Proceedings*, CEUR-WS.org, pp. 219–224.

[23] HARRIS, C. R., MILLMAN, K. J., VAN DER WALT, S. J., GOMMERS, R., VIRTANEN, P., COURNAPEAU, D., WIESER, E., TAYLOR, J., BERG, S., SMITH, N. J., KERN, R., PICUS, M., HOYER, S., VAN KERKWIJK, M. H., BRETT, M., HALDANE, A., FERNÁNDEZ DEL RÍO, J., WIEBE, M., PETERSON, P., GÉRARD-MARCHANT, P., SHEPPARD, K., REDDY, T., WECKESSER, W., ABBASI, H., GOHLKE, C., AND OLIPHANT, T. E. Array programming with NumPy. *Nature 585* (2020), 357–362.

[24] HOLOHAN, N., BRAGHIN, S., AONGHUSA, P. M., AND LEVACHER, K. Diffprivlib: The ibm differential privacy library, 2019.

[25] JAGIELSKI, M., ULLMAN, J. R., AND OPREA, A. Auditing differentially private machine learning: How private is private sgd? In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual* (2020).

[9] https://doi.org/10.5281/zenodo.15599462

[26] JIANG, H. Uniform convergence rates for kernel density estimation. In *Proceedings of the 34th International Conference on Machine Learning (ICML)* (2017).

[27] JOHNSON, M. Fix prng key reuse in differential privacy example, 2023. GitHub Pull Request #3646, [Accessed 08-Jan-2024].

[28] KLAUS, B., AND STRIMMER, K. *fdrtool: Estimation of (Local) False Discovery Rates and Higher Criticism*, 2024. R package version 1.2.18.

[29] KONG, W., MUÑOZ MEDINA, A., RIBERO, M., AND SYED, U. Dp-auditorium: A large-scale library for auditing differential privacy. In *IEEE Symposium on Security and Privacy, SP 2024, San Francisco, CA, USA, May 19-23, 2024* (2024), IEEE, pp. 110–126.

[30] KOSKELA, A., AND MOHAMMADI, J. Auditing differential privacy guarantees using density estimation. *arXiv preprint 2406.04827v3* (2024).

[31] KUTTA, T., ASKIN, Ö., AND DUNSCHE, M. Lower bounds for rényi differential privacy in a black-box setting. In *IEEE Symposium on Security and Privacy, SP, San Francisco, CA, USA, May 19-23* (2024).

[32] LIU, X., AND OH, S. Minimax optimal estimation of approximate differential privacy on neighboring databases. In *NeurIPS'19* (2019).

[33] LOKNA, J., PARADIS, A., DIMITROV, D. I., AND VECHEV, M. T. Group and attack: Auditing differential privacy. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security (CCS)* (2023).

[34] LU, Y., MAGDON-ISMAIL, M., WEI, Y., AND ZIKAS, V. Eureka: A general framework for black-box differential privacy estimators. In *SP'24* (2024).

[35] LYU, M., SU, D., AND LI, N. Understanding the sparse vector technique for differential privacy. *Proceedings of the VLDB Endowment 10*, 6 (2017).

[36] MAHLOUJIFAR, S., MELIS, L., AND CHAUDHURI, K. Auditing f-differential privacy in one run. *arXiv preprint arXiv:2410.22235* (2024).

[37] MIRONOV, I. On significance of the least significant bits for differential privacy. In *the ACM Conference on Computer and Communications Security (CCS)* (2012).

[38] NASR, M., HAYES, J., STEINKE, T., BALLE, B., TRAMÈR, F., JAGIELSKI, M., CARLINI, N., AND TERZIS, A. Tight auditing of differentially private machine learning. In *32nd USENIX Security Symposium (USENIX Security 23)* (2023).

[39] NEYMAN, J., AND PEARSON, E. S. 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 (1933), 289–337.

[40] PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., ET AL. Scikit-learn: Machine learning in python. *Journal of machine learning research 12*, Oct (2011), 2825–2830.

[41] SCOTT, D. W. *Multivariate Density Estimation: Theory, Practice, and Visualization*, 2nd ed. Wiley Series in Probability and Statistics. Wiley, 2015.

[42] SHEATHER, S. J., AND JONES, M. C. A reliable data-based bandwidth selection method for kernel density estimation. *Journal of the Royal Statistical Society. Series B (Methodological) 53*, 3 (1991), 683–690.

[43] STEINKE, T., NASR, M., AND JAGIELSKI, M. Privacy auditing with one (1) training run. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023* (2023).

[44] TSCHANTZ, M. C., KAYNAR, D. K., AND DATTA, A. Formal verification of differential privacy for interactive systems (extended abstract). In *Twenty-seventh Conference on the Mathematical Foundations of Programming Semantics (MFPS)* (2011), Electronic Notes in Theoretical Computer Science.

[45] VAN DER VAART, A. W., AND WELLNER, J. A. *Weak Convergence and Empirical Processes. With Applications to Statistics*. Springer Series in Statistics., New York, 1996.

[46] VIRTANEN, P., GOMMERS, R., OLIPHANT, T. E., HABERLAND, M., REDDY, T., COURNAPEAU, D., BUROVSKI, E., PETERSON, P., WECKESSER, W., BRIGHT, J., VAN DER WALT, S. J., BRETT, M., WILSON, J., MILLMAN, K. J., MAYOROV, N., NELSON, A. R. J., JONES, E., KERN, R., LARSON, E., CAREY, C. J., POLAT, İ., FENG, Y., MOORE, E. W., VANDERPLAS, J., LAXALDE, D., PERKTOLD, J., CIMRMAN, R., HENRIKSEN, I., QUINTERO, E. A., HARRIS, C. R., ARCHIBALD, A. M., RIBEIRO, A. H., PEDREGOSA, F., VAN MULBREGT, P., AND SCIPY 1.0 CONTRIBUTORS. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods 17* (2020), 261–272.

[47] WAND, M. *KernSmooth: Functions for Kernel Smoothing Supporting Wand & Jones (1995)*, 2025. R package version 2.23-26.

[48] WANG, Y., DING, Z., KIFER, D., AND ZHANG, D. Checkdp: An automated and integrated approach for proving differential privacy or finding precise counterexamples. In *CCS'20* (2020).

[49] WANG, Y., DING, Z., WANG, G., KIFER, D., AND ZHANG, D. Proving differential privacy with shadow execution. In *Proceedings of the 40th ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI)* (2019).

[50] YOUSEFPOUR, A., SHILOV, I., SABLAYROLLES, A., TESTUGGINE, D., PRASAD, K., MALEK, M., NGUYEN, J., GHOSH, S., BHARADWAJ, A., ZHAO, J., ET AL. Opacus: User-friendly differential privacy library in pytorch. *arXiv preprint arXiv:2109.12298* (2021).

[51] ZANELLA-BÉGUELIN, S., WUTSCHITZ, L., TOPLE, S., SALEM, A., RÜHLE, V., PAVERD, A., NASERI, M., KÖPF, B., AND JONES, D. Bayesian estimation of differential privacy. In *Proceedings of the 40th International Conference on Machine Learning (ICML)* (2023).

[52] ZHANG, D., AND KIFER, D. Lightdp: towards automating differential privacy proofs. In *Proceedings of the 44th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL)* (2017).

A Appendix

The appendix is dedicated to the technical details of our results. The proofs can be found in an extended arxiv version.

B Additional Experiments and Details

In this section, we provide some additional details on our experiments and implementations.

Table 1: Overview of Notation Used in the Paper

Notation	Description
D, D'	Pair of adjacent databases
M	(f-)DP Mechanism
$\Pr[], \mathbb{E}[]$	Probability, Expectation
P, Q	Output distributions of $M(D), M(D')$
$[P]_\eta$	Mixture distribution with parameter η
p, q	Probability densities of P, Q
α, β	type-I & type-II errors (typically of the Neyman-Pearson test)
$\hat{\alpha}_h, \hat{\beta}_h$	Estimated errors using KDE
$\tilde{\alpha}, \tilde{\beta}$	Estimated errors using k-NN (typically of the Neyman-Pearson test)
T	optimal trade-off curve for p, q
$T^{(0)}$	trade-off curve that is audited
T_h	trade-off curve of perturbed LR test
$\hat{T}_h$	estimated trade-off curve using KDE
η	threshold in LR tests
	vulnerability
$\hat{\eta}^*$	estimated threshold of maximum vulnerability
λ	randomization parameter in Neyman-Pearson test
h	randomization parameter in perturbed LR test
$\phi, \phi_{k,n}^{\mathtt{NN}}$	generic classifier, k-NN classifier
ϕ^*	Bayes optimal classifier
$\gamma, w(\gamma)$	confidence level & margin of error
$\square_\gamma$	confidence region for type-I-type-II errors
n, n_1, n_2	sample size parameters

Algorithm 3 PLRT: A Perturbed Likelihood Ratio Test Algorithm for f-DP Estimation

```
Require: Black-box access to $M$; Threshold $\eta > 0$; Sample size $n$,
         databases $D, D'$.
Ensure: An estimate $(\hat{\alpha}(\eta), \hat{\beta}(\eta))$ of $(\alpha(\eta), \beta(\eta))$ for tuple $(P, Q)$,
where $M(D)$ and $M(D')$ are distributed according to $P, Q$, respectively.
Choose perturbation parameter $h$.
Set the density estimation algorithm $\mathcal{A}$. By default, use the KDE algorithm.
function PLRT Estimator $\mathtt{PLRT}_{\mathcal{A}}^{h}(M, D, D', \eta, n)$
    Compute the estimated densities $\hat{p}$ and $\hat{q}$ by running $\mathcal{A}$ on $n$ independent copys of $M(D)$ and $M(D')$, respectively.
    Compute $\hat{\alpha}(\eta) \leftarrow \int_{x \in [-h/2, h/2]} \frac{1}{h} \int_{\hat{q}/\hat{p} > \eta + x} \hat{p}$
    Compute $\hat{\beta}(\eta) \leftarrow 1 - \int_{x \in [-h/2, h/2]} \frac{1}{h} \int_{\hat{q}/\hat{p} > \eta + x} \hat{q}$
    Return $(\hat{\alpha}(\eta), \hat{\beta}(\eta))$
end function
```

B.1 Implementation details

Algorithm 3 gives a pseudo-code of our trade-off curve estimator $\hat{T}_h$, presented in Section 4.

Next, we turn to the DP-SGD algorithm from our Experiments section. The pseudocode for that algorithm can be found in Algorithm 4 below. Note that we add Gaussian noise $Z_t \sim \mathcal{N}(0, \sigma^2)$ to the parameter θ_t at each iteration of DP-SGD. The operator Π_Θ projects the averaged and perturbed gradient onto the space Θ and is thus similar to clipping that gradient. We can derive the exact trade-off function of this algorithm for our choice of databases in (10) and our specifications from Section 6.1. More concretely, we first consider the distribution of DP-SGD on $D = (0, \ldots, 0)$ and note that

$$\theta_{t+1} = \theta_t - \rho\,(\theta_t + Z_{t+1})$$

for each $t \in \{0, \ldots, \tau\}$. Some calculations then yield that $\Theta_\tau \sim \mathcal{N}(0, \bar{\sigma}^2)$ with

$$\bar{\sigma}^2 = \rho^2 \sigma^2 \, \frac{1 - (1-\rho)^{2\tau}}{1 - (1-\rho)^2}. \tag{13}$$

Similarly, we have for $D' = (1, 0, \ldots, 0)$ that

$$\theta_{t+1} = (1-\rho)\,\theta_t + \rho\, Z_{t+1}$$

for each $t \in \{0, \ldots, \tau\}$. Here, Z_t is a Gaussian mixture with

$$Z_t \sim \frac{1}{2}\mathcal{N}(0, \sigma^2) + \frac{1}{2}\mathcal{N}\left(\frac{1}{m}, \sigma^2\right).$$

We can then see that $\theta_\tau = \tilde{Z}_1 + \cdots + \tilde{Z}_\tau$ where the $\tilde{Z}_t$ are independent Gaussian mixtures with

$$\tilde{Z}_t \sim \frac{1}{2}\mathcal{N}\left(0, \rho^2\,(1-\rho)^{2(\tau-t)}\,\sigma^2\right) + \frac{1}{2}\mathcal{N}\left(\frac{\rho(1-\rho)^{\tau-t}}{m}, \rho^2\,(1-\rho)^{2(\tau-t)}\,\sigma^2\right).$$

By defining

$$\mu_I := \sum_{t \in I} \frac{\rho(1-\rho)^{\tau-t}}{m} \tag{14}$$

and choosing $\bar{\sigma}$ as in (13), we get that

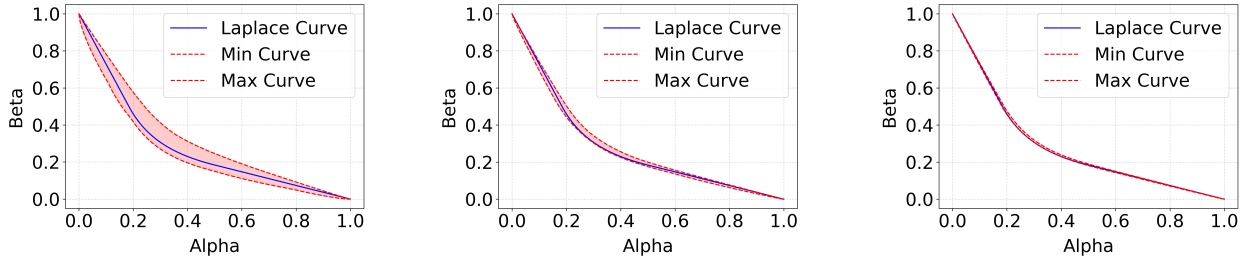
$$\theta_\tau \sim \sum_{t \subset \{1, \ldots, \tau\}} \frac{1}{2^\tau} \mathcal{N}(\mu_I, \bar{\sigma}^2).$$

Having derived the distribution of $M(D)$ and $M(D')$, we take a look at the corresponding LR-test g and note that it can be expressed as

$$g(x) = \begin{cases} 1 & x > c \\ 0 & x \leq c \end{cases}$$

for some threshold c. A few calculations then yield the trade-off curve

$$T_{SGD}(\alpha) = \sum_{I \subset \{1, \ldots, \tau\}} \frac{1}{2^\tau} \Phi\left(\Phi^{-1}(1-\alpha) - \frac{\mu_I}{\bar{\sigma}}\right).$$

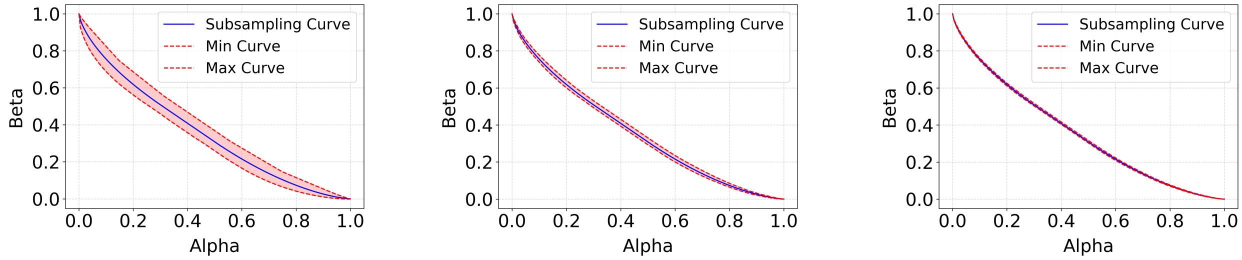


(a) $n_1 = 10^3$

(b) $n_1 = 10^4$

(c) $n_1 = 10^5$

Figure 9: Estimation of the Laplace Trade-off curve T_{Lap} for varying sample sizes. Min- and Max Curve lower- and upper bound the worst point-wise deviation from the true curve T_{Lap} over 1000 simulations.



(a) $n_1 = 10^3$

(b) $n_1 = 10^4$

(c) $n_1 = 10^5$

Figure 10: Estimation of the Subsampling Trade-off curve T_{Sub} with the Gaussian mechanism for $\mu = 1$ and varying sample sizes. Min- and Max Curve lower- and upper bound the worst point-wise deviation from the true curve T_{Sub} over 1000 simulations.

Algorithm 4 DP-SGD Algorithm

```
Require: Dataset $x = (x_1, \dots, x_r)$, loss function $\ell(\theta, x)$,
         Parameters: initial state $\theta_0$, learning rate $\rho$, batch size $m$,
         time horizon $\tau$, noise scale $\sigma$, closed and convex space $\Theta$.
Ensure: Final parameter $\theta_\tau$.
for $t = 1, \dots, \tau$ do
    Subsampling: Take a uniformly random subsample $I_t \subseteq \{1, \dots, r\}$ with batch size $m$.
    for $i \in I_t$ do
        Compute gradient: $v_t^{(i)} \leftarrow \nabla_\theta \ell(\theta_t, x_i)$
    end for
    Average, perturb, and descend:
        $$\theta_{t+1} \leftarrow \theta_t - \rho\, \Pi_\Theta \left( \frac{1}{m} \sum_{i \in I_t} v_t^{(i)} + Z_t \right)$$
end for
Output: $\theta_\tau$
```

B.2 Additional Algorithms

We test our estimation procedure on the Laplace and Subsampling algorithm, which often serve as building blocks in more sophisticated privacy mechanisms. We select the same setting for our experiments as in Section 6 and choose D and D' as in (10).

Laplace mechanism. We consider the summary statistic $S(x) = \sum_{i=1}^{10} x_i$ and the mechanism

$$M(x) := S(x) + Y \ ,$$

where $Y \sim \mathcal{Lap}(0, \sigma)$. The statistic $S(x)$ is privatized by the random noise Y if the scale parameter $\sigma > 0$ of the Laplace distribution is chosen appropriately. We choose $\sigma = 1$ for our experiments and observe that the optimal trade-off curve is given by

$$T_{Lap}(\alpha) = \begin{cases} 1 - e\alpha, & \alpha < e^{-1}/2 \ , \\ e^{-1}/4\alpha, & e^{-1}/2 \leq \alpha \leq 1/2 \ , \\ e^{-1}(1-\alpha), & \alpha > 1/2. \end{cases}$$

We point the interested reader to [19] for more details on how to derive T_{Lap}.

Subsampling algorithm. Random subsampling provides an effective way to enhance the privacy of a DP mechanism M. We only provide a rough outline here and refer for details to [19]. In simple words, we choose an integer m with $1 \leq m < r$, where r is the size of the database D. We then draw a random subsample of size m from D, giving us the smaller database $\bar{D}$ of size m. The mechanism M is then applied to $\bar{D}$ instead of D, providing users with an additional layer of privacy (if a user is not part of $\bar{D}$, their privacy cannot be compromised). The amplifying effect that subsampling has on privacy is visible in the optimal trade-off curve: If M has the trade-off curve T, then $M(\bar{D})$ has the trade-off curve

$$\bar{T}(\alpha) = \frac{m}{r}T(\alpha) + \frac{r-m}{r}(1-\alpha),$$

which is strictly more private than T for any $m < r$. A minor technical peculiarity of subsampling is that the resulting curve $\bar{T}$ is not necessarily symmetric, even if T is (see [19] for details on the symmetry of trade-off functions). Trade-off curves are usually considered to be symmetric and one can symmetrize $\bar{T}$ by applying a symmetrizing operator $\mathbf{C}$ with

$$\mathbf{C}[T](x) = \begin{cases} T(x), & x \in [0, x^*] \\ x^* + T(x^*) - x, & x \in [x^*, T(x^*)] \\ T^{-1}(x), & x \in [T(x^*), 1], \end{cases}$$

where x^* is the unique fix-point of T with $T(x^*) = x^*$ (for more details we refer to [19]). Another mathematical representation of $\mathbf{C}$ that we use in our code is $\mathbf{C}(T) = \min\{T, T^{-1}\}^{**}$, where the index $**$ signifies double convex conjugation. We incorporate this operation into our estimation procedure by simply applying $\mathbf{C}$ to our estimate of the trade-off function T. For our experiments involving subsampling, we use the Gaussian mechanism for M (with $\sigma = 1$) and obtain the subsampled version M' by fixing the parameter $m = 5$ (recall that $r = 10$).

Similar to the experiments section, we construct figures that upper and lower bound the worst case errors for the Laplace mechanism and the Subsampling algorithm over 1000 simulation runs. We can see again that the error of the estimator $\hat{T}_h$ shrinks significantly, as n_1 grows.

B.3 Comparison to credible intervals

In this work, we consider the construction of confidence bounds as common in frequentist statistics. If γ is set to 1% in Theorem 5.2, this means that (on average) in 100 audits of a correct algorithm only at most one violation will be (erroneously) detected. These kinds of guarantees are the gold standard in empirical sciences and we believe they are the guarantees real users would care about. It is worth noting that there exist other types of statistical results, including credible results from Bayesian statistics such as by [51], who work on approximate DP. It is important to point out that Bayesian results are very different from frequentist approaches. One difference is their performance, because credible intervals do not generally provide the same bounds on false detection rates as frequentist results. We illustrate this point with a minimal simulation. The aim is to make a confidence/credible interval for the bias of a coin. We simulate $n = 500$ coin flips per trial for $k = 10^5$ trials, with varying bias (p). Frequentist confidence intervals use the sample proportion and normal approximation, while Bayesian credible intervals rely on a standard Beta prior ($\alpha = 10, \beta = 10$) updated with observed data. Coverage is assessed by checking if intervals contain the true p. The targeted confidence/credible is $1-\gamma$ with $\gamma = 0.1$ and results are displayed in Figure 11. As we can see, frequentist intervals hold repeated-sampling guarantees, while Bayesian credible intervals depend on priors and lack such guarantees under repeated sampling.

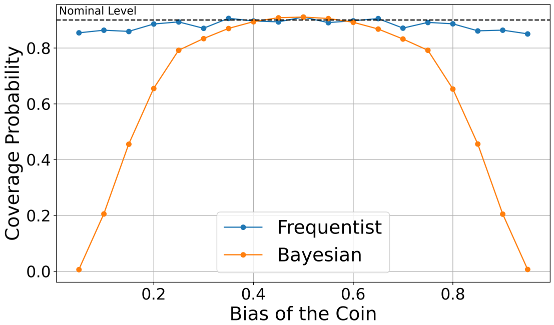


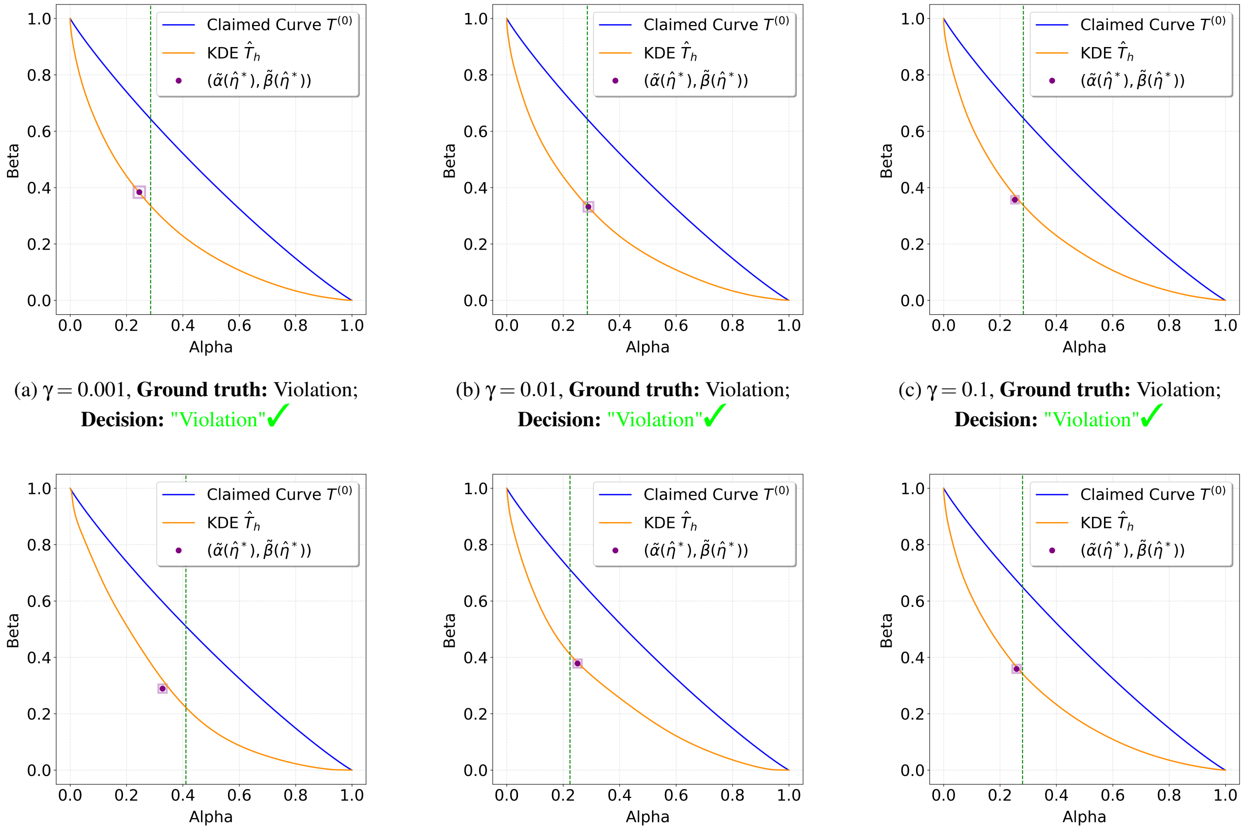
Figure 11: Empirical coverage probabilities for frequentist and Bayesian intervals based on 10^5 simulation runs.

B.4 Additional simulations

We present some results that complement the main findings in our experiment section. We use the same setup as described in our experiments and investigate a faulty implementation of the Gaussian mechanism. We study two things: First, the impact of the parameter γ, where we vary γ between very small and relatively large values. As we can see, smaller values of γ lead to larger boxes $\square_\gamma$ which make it harder for the auditor to detect violations. Secondly, we consider the impact of the sample size n_1 ranging from the very modest value of 10^2 up to 10^4. We see that the sample size has very little impact on the performance of the procedure and it already works well for fairly small samples n_1 (n_2 has a greater impact, as we have seen in our experiments). Finally, we have also reported the runtimes for different mechanism in Table 2 and Table 3.

B.5 Proofs for Goal 1 (Estimation)

Consequences of Theorem 4.2 The main result in Section 4 is Theorem 4.2. Lemma 4.1 can be seen as a special case, putting $\hat{p} = p, \hat{q} = q$. Then, Assumption 2 is met for the constant sequence $a_n = 0$. It follows by this construction that $\hat{T}_h = T_h$, is non-random and only depends



(a) $\gamma = 0.001$, **Ground truth:** Violation; **Decision:** "Violation"✓

(b) $\gamma = 0.01$, **Ground truth:** Violation; **Decision:** "Violation"✓

(c) $\gamma = 0.1$, **Ground truth:** Violation; **Decision:** "Violation"✓

(d) $n_1 = 10^2$, **Ground truth:** Violation; **Decision:** "Violation"✓

(e) $n_1 = 10^3$, **Ground truth:** Violation; **Decision:** "Violation"✓

(f) $n_1 = 10^4$, **Ground truth:** Violation; **Decision:** "Violation"✓

Figure 12: **Auditing a faulty Mechanism:** Claimed Curve $T^{(0)} = T_{Gauss}$ with $\mu = 0.2$, but in reality $\mu = 1$. For (a),(b),(c) we consider $n_1 = 10^4$, and for (d),(e),(f) we have considered various sample sizes for the KDEs, respectively. Throughout the simulations we keep $n_2 = 10^4$ fixed, and confidence intervals in (d), (e), and (f) are computed with level $\gamma = 0.05$

Algorithm	**Runtime in seconds**
Gaussian mechanism	26.3
Laplace mechanism	30.51
Subsampling mechanism	27.82
DP-SGD	61.1

Table 2: Average runtimes of Algorithm 3 for $n_1 = 10^5$ over 10 runs to obtain the full trade-off curve T.

Algorithm	**Runtime in seconds**
Gaussian mechanism	62.63
Laplace mechanism	67.04
Subsampling mechanism	66.98
DP-SGD	114.86

Table 3: Average runtimes of Algorithm 1 for $n_2 = 10^6$ over 5 runs to obtain one point of the trade-off curve T with confidence region.

on h. Any choice of $h \downarrow 0$ is permissible and Lemma 4.1 follows from the Theorem. Proposition 4.3 too is a direct consequence of Theorem 4.2. To see this, we notice that

$$\begin{aligned} & T_0(\hat{\alpha}_h(\hat{\eta}^*)) - T(\hat{\alpha}_h(\hat{\eta}^*)) \\ = & T_0(\hat{\alpha}_h(\hat{\eta}^*)) - \hat{T}_h(\hat{\alpha}_h(\hat{\eta}^*)) + o_P(1) \\ = & \sup_{\alpha \in [0,1]} \{T_0(\alpha) - \hat{T}_h(\alpha)\} + o_P(1) \\ = & \sup_{\alpha \in [0,1]} \{T_0(\alpha) - T(\alpha)\} + o_P(1). \end{aligned}$$

In the first and last step, we have used the uniform convergence of Theorem 4.2, which allows us to replace T by $\hat{T}_h$ while only incurring an $o_P(1)$ error. In the second step, we have used the definition of $\hat{\alpha}_h(\hat{\eta}^*)$ as the maximizer of the difference between T_0 and $\hat{T}_h$. Thus Proposition 4.3 follows. We now turn to the proof of the theorem. The proof is presented for densities on the real line. Extensions to $\mathbb{R}^d$ are straightforward and therefore not discussed.

Preliminaries Recall that a complete separable metric space is Polish. The real numbers, equipped with the absolute value distance is a Polish space. The continuous functions $\mathcal{C}_0$ on the real line that vanish at $\pm\infty$, i.e. that satisfy

$$\lim_{x\to\infty} f(x) = \lim_{x\to\infty} f(-x) = 0 \qquad (15)$$

is a Polish space if equipped with the supremum norm

$$\|f\| := \sup_{x\in\mathbb{R}} |f(x)|.$$

The product of complete, separable metric spaces is complete and separable if equipped with the maximum metric, i.e. the space $\mathcal{C}_0 \times \mathcal{C}_0 \times \mathbb{R} \times \mathbb{R}$ is Polish. Now, the vector

$$(\hat{p}, \hat{q}, \|\hat{p} - p\|_\infty / a_n, \|\hat{q} - q\|_\infty / a_n)$$

lives on this space (for each n) and convergences to the limit $(p, q, 0, 0)$ in probability (see Assumption 2). Accordingly we can use Skorohod's theorem to find a probability space, where this convergence is a.s.

$$(\hat{p}, \hat{q}, \|\hat{p} - p\|_\infty / a_n, \|\hat{q} - q\|_\infty / a_n) \to (p, q, 0, 0) \quad a.s.$$

It is a direct consequence that on this space it holds a.s.

$$\|\hat{p} - p\| = o(a_n), \quad \|\hat{q} - q\| = o(a_n).$$

In the following, we will work on this modified probability space and exploit the a.s. convergence. We will fix the outcome and regard $\hat{p}, \hat{q}$ as sequences of deterministic functions, converging to their respective limits at a rate $o(a_n)$.

Next, it suffices to show the desired result pointwise for any α. This reduction is well-known. For a sequence of continuous, monotonically decreasing functions $(f_n)_n$ living on the unit interval $[0,1]$, pointwise convergence to a continuous, monotonically decreasing limit f on $[0,1]$ implies uniform convergence. The same argument lies at the heart of the proof of the famous Glivenko-Cantelli Theorem (see [45]). We now want to demonstrate the convergence $|\hat{T}(\alpha) - T(\alpha)| = o(1)$ pointwise. More precisely, we will demonstrate that for the pair $(\alpha, T(\alpha))$, there exist values of η such that $\hat{\alpha}_h(\eta) \to \alpha$ and $\hat{\beta}_h(\eta) \to T(\alpha)$. Since the proofs of both convergence results work exactly in the same way, we restrict ourselves in this proof to show that $\hat{\alpha}_h(\eta) \to \alpha$. So let us consider a fixed but arbitrary value of $\alpha \in [0,1]$ and begin the proof.

Case 1: We first consider the case where $\eta \geq 0$ (the threshold in the optimal LR test) is such that the set $\{q/p = \eta\}$ has 0 mass. In this case, the coin toss with probability λ can be ignored (it happens with probability 0) and we can define the type-I-error α of the Neyman-Pearson test as

$$\alpha = \int p \cdot \mathbb{I}\{q/p > \eta\}.$$

In this case, we want to show that

$$\begin{aligned}
&\int_{x\in[-h/2,h/2]} \frac{1}{h} \int_{\hat{q}/\hat{p} > \eta + x} \hat{p} \\
&= \int \int_{x\in[-h/2,h/2]} \hat{p} \frac{1}{h} \mathbb{I}\{\hat{q}/\hat{p} > \eta + x\} =: \int \hat{g} \\
&\to \int_{q/p > \eta} p = \int p \cdot \mathbb{I}\{q/p > \eta\} =: \int g.
\end{aligned}$$

Here we have defined the functions $g, \hat{g}$ in the obvious way. We will now show $\hat{g}$ converges pointwise to g. For this purpose consider the interval $[-K, K]$ for a large enough K, such that

$$\int_{[-K,K]^c} p < \zeta \qquad \text{and} \qquad \int_{[-K,K]^c} q < \zeta$$

for a number ζ that we can make arbitrarily small. Given the uniform convergence of the density estimators on the interval $[-K, K]$ it holds for all n sufficiently large that also

$$\int_{[-K,K]^c} \hat{p} < \zeta \qquad \text{and} \qquad \int_{[-K,K]^c} \hat{q} < \zeta.$$

Accordingly we have

$$\left| \int \hat{g} - g \right| \leq 2\zeta + \left| \int_{[-K,K]} \hat{g} - g \right|.$$

We then focus on the second term on the right and fix some argument $y \in [-K, K]$. It holds that either $q(y)/p(y)$ is bigger or smaller than η (equality occurs only on a null-set and can therefore be neglected). Let us focus on the case where $q(y)/p(y) > \eta$. If this is so, then it follows that in a small environment, say for $y' \in [y - \zeta', y + \zeta']$ we also have $q(y')/p(y') > \eta$. For all large enough n it follows that $h/2 < \zeta'$. Then, it is easy to see that also $\hat{q}(y')/\hat{p}(y') > \eta$ for all $y' \in [y - \zeta', y + \zeta']$ simultaneously, for all sufficiently large n. If this is the case, the indicators in the definition of $\hat{g}, g$ become 1 and $\hat{g} = \hat{p}$, $g = p$. So, we have pointwise $\hat{g}(y) = \hat{p}(y) \to p(y) = g(y)$. Since $\hat{g}$ is also bounded for all sufficiently large n (since the integral over the indicator is bounded and the sequence

$\hat{p}$ is uniformly convergent to the bounded function p) we obtain by the theorem of dominated convergence that

$$\left| \int_{[-K,K]} \hat{g} - g \right| \to 0.$$

This shows that

$$\limsup_n |\hat{\alpha}_h(\eta) - \alpha| = O(\zeta).$$

Finally, letting $\zeta \downarrow 0$ in a second limit shows the desired approximation in this case.

Case 2: Next, we consider the case where the set $\{q/p = \eta\}$ has positive mass for some $\eta > 0$.[10] This means that the coin-flip in the definition of the optimal LR test plays a role and we set the probability λ to some value in $[0,1]$. We then consider as estimator the value $\hat{\alpha}(\eta - bh)$ for a value b that we will determine below. Let us, for ease of notation, define the probability

$$L := \int_{q/p=\eta} p$$

and appreciate that then

$$\alpha = \alpha' + O(\zeta) + \lambda L. \tag{16}$$

We explain the decomposition: In equation (16), α' is the rejection probability of the LR test defined by the decision to reject whenever $q(y)/p(y) > \eta + \zeta''$ for some small number ζ''. For all small enough values of ζ'' the threshold $\eta + \zeta''$ is not a plateau value (there are only finitely many of them; see Assumption 1). It follows that

$$\alpha' = \int p \cdot \mathbb{I}\{q/p > \eta + \zeta''\}.$$

Next, for any fixed constant $\zeta > 0$ we can choose ζ'' small enough such that

$$\int p \cdot \mathbb{I}\{\eta < q/p \leq \eta + \zeta''\} < \zeta. \tag{17}$$

This explains the second term on the right of equation (16). The third term corresponds to the probability of rejecting whenever $q/p = \eta$ (this probability is L) times the probability that the coin shows heads (reject) with probability λ.

Now, using these definitions, we decompose the set

$$\begin{aligned} &\{\hat{q}/\hat{p} > \eta - bh + x\} \\ &= \{\eta + \zeta'' \geq \hat{q}/\hat{p} > \eta - bh + x\} \cup \{\hat{q}/\hat{p} > \eta + \zeta''\}. \end{aligned}$$

This yields the decomposition

$$\begin{aligned} &\hat{\alpha}_h(\eta - bh) = \hat{\alpha}_h(\eta + \zeta'') \qquad (18) \\ &+ \int \int_{x \in [-h/2, h/2]} \hat{p} \, \frac{1}{h} \, \mathbb{I}\{\eta + \zeta'' \geq \hat{q}/\hat{p} > \eta - bh + x\}. \end{aligned}$$

Now, by part 1 of this proof we have

$$|\hat{\alpha}_h(\eta + \zeta'') - \alpha'| = o(1).$$

Next, we study the integral on the right side of eq. (18) and for this purpose define the objects

$$\begin{aligned} \tilde{g} &:= \int_{x \in [-h/2, h/2]} \hat{p} \, \frac{1}{h} \, \mathbb{I}\{A_1\}, \\ \tilde{f} &:= \int_{x \in [-h/2, h/2]} \hat{p} \, \frac{1}{h} \, \mathbb{I}\{A_2\}. \\ A_1 &:= \{\eta + \zeta'' \geq \hat{q}/\hat{p} > \eta - bh + x, q/p = \eta\}, \\ A_2 &:= \{\eta + \zeta'' \geq \hat{q}/\hat{p} > \eta - bh + x, q/p \neq \eta\}. \end{aligned}$$

Now, let us consider a value y where $q(y)/p(y) \neq \eta$ and for sake of argument let us focus on the (more difficult) case $q(y)/p(y) > \eta$. If $q(y)/p(y) > \eta + \zeta''$, it follows that eventually $\hat{p}(y)/\hat{q}(y) > \eta + \zeta''$ and hence $\tilde{f}(y) = 0$. The case where $q(y)/p(y) = \eta + \zeta''$ is a null-set and hence negligible (it is not a plateau value). The case where $q(y)/p(y) \in (\eta, \eta + \zeta'')$ implies that eventually $\hat{p}(y)/\hat{q}(y) \in (\eta, \eta + \eta'')$ and thus eventually $\tilde{f}(y) = \hat{p}(y)$ which converges pointwise to p. Thus, we have by dominated convergence that

$$\int \tilde{f} \to \int p \cdot \mathbb{I}\{\eta < q/p \leq \eta + \zeta''\} < \zeta.$$

The fact that the integral is bounded by ζ was established in eq. (17). This means that for all n large enough we have

$$\int \tilde{f} < \zeta.$$

Now, let us focus on a value of y where $q(y)/p(y) = \eta$. In this case it follows that $q(y), p(y) > 0$ and we have

$$\frac{\hat{q}(y)}{\hat{p}(y)} = \frac{q(y)}{p(y)} + o(a_n) = \eta + o(a_n).$$

Notice that we can rewrite $\tilde{g}$ as

$$\int_{x \in [-1/2, 1/2]} \hat{p} \, \mathbb{I}\{\eta + \zeta'' \geq \hat{q}/\hat{p} > \eta - bh + hx, q/p = \eta\}.$$

Now, for any $x > b$ it follows that the indicator will eventually be 0, because

$$\hat{q}/\hat{p} = \eta + o(a_n) << \eta + h(x - b)$$

(because $a_n = o(h)$ by assumption in the Theorem). By similar reasoning the indicator is 1 if $x < b$. This means that $\tilde{g}$ converges for any fixed y with $q(y)/p(y) = \eta$ to $p(y) \cdot (1/2 + b)$ and using majorized convergence yields

$$\int \tilde{g} \to (1/2 + b) \int_{q/p=\eta} p = (1/2 + b) L.$$

[10] We omit the simpler case where $\eta = 0$ and $L = 0$ anyways.

Now, we can choose $b = \lambda - 1/2$ to get that the right side is equal to λL. Putting these considerations together, we have shown that

$$\limsup_{n} |\alpha - \hat{\alpha}_h(\eta - [\lambda - 1/2]h)| = O(\zeta).$$

Taking the limit $\zeta \downarrow 0$ afterwards yields the desired result.

B.6 Proofs for Goal 2 (Auditing)

Before we proceed to the proofs, we state a simple but useful consequence of the Neyman-Pearson Lemma.

Corollary B.1 *Let set $\mathcal{S}_\eta = \{x : p(x)/q(x) \leq \eta\}$. For $\alpha \in [0,1]$, if there exists η such that $\Pr_{X\sim P}[X \in \mathcal{S}_\eta] = \alpha$, then it holds that*

$$\beta(\alpha) = 1 - \Pr_{X\sim Q}[X \in \mathcal{S}_\eta].$$

As a next step, we prove a theoretical result connecting the output of the BayBox estimator for the theoretical (in practice unknown) Bayes classifier ϕ^*.

Lemma B.2 *Let η, $(\alpha(\eta), \beta(\eta))$, $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$, and ϕ be as defined in Algorithm 1. Set ϕ to the Bayes optimal classifier ϕ^* for the corresponding Bayesian classification problem. Then, with probability $1-\gamma$,*

$$|\tilde{\alpha}(\eta) - \alpha(\eta)| \leq \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}}$$

$$\left|\tilde{\beta}(\eta) - \beta(\eta)\right| \leq \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}}.$$

Proof. [**Proof of Lemma B.2**] We prove the statement that $|\tilde{\alpha}(\eta) - \alpha(\eta)| \leq \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}}$ if $\eta \geq 1$ with probability $\geq 1 - \gamma/2$. The proof of the second statement follows a similar approach. We begin with a few definitions. Let the observation set be defined as

$$O := \texttt{Supp}(P) \cup \texttt{Supp}(Q) \cup \{\bot\},$$

i.e. the range of observation. Define the indicator function $\mathbb{I}_{\mathcal{S}_\eta} : O \mapsto \{0,1\}$, which takes as input an observation x from the observation set O, outputting 1 if $x \in \mathcal{S}_\eta$ and 0 otherwise. Also, recall the definition of the set $\mathcal{S}_\eta = \{x : p(x)/q(x) \leq \eta\}$ as the set of all observation $x \in O$ where $p(x)$ is less than or equal to $\eta q(x)$ (as before p, q are the densities of distributions P, Q).

We first show that $\mathbb{I}_{\mathcal{S}_\eta}$ is exactly the Bayes classifier ϕ^* for the Bayesian binary classification problem $\mathbf{P}\left[[P]_\eta, Q\right]$. We prove this by showing for every $x \in O$, $\phi^*(x) = \mathbb{I}_{\mathcal{S}_\eta}(x)$. Therefore, consider the tuple of random variable $(X,Y) \sim \mathbf{P}\left[[P]_\eta, Q\right]$. Then, for every observation $x \in O \setminus \{\bot\}$, we have

$$\begin{aligned}
\phi^*(x) &= \operatorname*{arg\,max}_{\{0,1\}}\{\Pr[Y=0|X=x], \Pr[Y=1|X=x]\} \\
&\qquad\qquad \text{(by Bayes classifier } \phi^*\text{'s construction)} \\
&= \operatorname*{arg\,max}_{\{0,1\}}\{\Pr[Y=0, X=x], \Pr[Y=1, X=x]\} \\
&\qquad\qquad \text{(by Bayes Theorem)} \\
&= \operatorname*{arg\,max}_{\{0,1\}}\{\frac{1}{\eta}p(x), q(x)\} \\
&= \mathbb{I}_{\mathcal{S}_\eta}(x). \qquad\qquad \text{(by } \mathbb{I}_{\mathcal{S}_\eta}\text{'s definition)}
\end{aligned}$$

For an observation $x = \bot$, it is easy to check $\phi^*(x) = \mathbb{I}_{\mathcal{S}_\eta}(x) = 0$, as $q(x) = 0$.

Then, we also observe that

$$\begin{aligned}
&\alpha(\eta) && (19)\\
&= \Pr_{X\sim P}[X \in \mathcal{S}_\eta] && \text{(By Corollary B.1)}\\
&= \Pr_{X\sim P}\left[\mathbb{I}_{\mathcal{S}_\eta}(X) = 1\right] \\
&= \Pr_{X\sim P}[\phi^*(X) = 1] && (\phi^* = \mathbb{I}_{\mathcal{S}_\eta})\\
&= \mathbb{E}_{X\sim P}[\phi^*(X)]
\end{aligned}$$

Recall that in algorithm 1, BayBox estimation $\mathtt{BB}^{\phi^*}$ computes the empirical mean of $\phi^*(X)$, i.e., $\tilde{\alpha}(\eta)$, as the estimate of $\alpha(\eta)$. By Hoeffding's Inequality, we finally conclude that

$$\begin{aligned}
&\Pr\left[|\tilde{\alpha}(\eta) - \alpha(\eta)| > \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}}\right] \\
&= \Pr\left[\left|\frac{1}{n}\sum_{i=1}^{n} Z_i - \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n} Z_i\right]\right| > \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}}\right] \\
&\qquad\qquad (Z_i \overset{\text{def}}{=} \phi^*(X_i), X_i \overset{\text{i.i.d.}}{\sim} P) \\
&\leq \gamma/2.
\end{aligned}$$

□

Remark 3 *For any classifier ϕ that is used as input of the BayBox algorithm, the output $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ will have a mean point $(\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta)))$ on or above the optimal trade-off curve. The reason is that $(\tilde{\alpha}(\eta), \tilde{\beta}(\eta))$ are the empirical type-I and type-II-errors of the test that rejects whenever an output is classified as belonging to D'. The means $\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta)$ correspond to the population version of the errors which by construction of the optimal trade-off curve are on or above it (no test has a better combination than the Neyman-Pearson test which demarcates the curve exactly).*

Proof. [**Proof of Theorem 5.1**] The proof of part 1) of the theorem follows in exact analogy to the proof of Lemma B.6 and we do not repeat it here. Now, we turn to the proof of part 2). Again we restrict ourselves to proving the statement about the type-I-errors $|\tilde{\alpha}(\eta)-\alpha(\eta)| \leq \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + \sqrt{\frac{144c_d^2}{n}\ln\frac{4}{\gamma}}$, and the statement on type-II-errors follows by a similar approach. With probability at least $1-\gamma/2$, we have

$$\begin{aligned}
&|\tilde{\alpha}(\eta)-\alpha(\eta)| \\
=& \left|\frac{1}{n}\sum_{i=1}^{n}\phi_{k,n}^{\text{NN}}(X_i) - \mathbb{E}\left[\frac{1}{n}\sum_{i=1}^{n}\phi^*(X_i)\right]\right| && (X_i \overset{\text{i.i.d.}}{\sim} P) \\
=& \left|\frac{1}{n}\sum_{i=1}^{n}\phi_{k,n}^{\text{NN}}(X_i) - \mathbb{E}\left[\phi^*(X)\right]\right| && (X \sim P) \\
\leq& \left|\frac{1}{n}\sum_{i=1}^{n}\phi_{k,n}^{\text{NN}}(X_i) - \mathbb{E}\left[\phi_{k,n}^{\text{NN}}(X)\right]\right| + \left|\mathbb{E}\left[\phi_{k,n}^{\text{NN}}(X)\right] - \mathbb{E}\left[\phi^*(X)\right]\right| \\
\leq& \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + \left|\mathbb{E}\left[\phi_{k,n}^{\text{NN}}(X)\right] - \mathbb{E}\left[\phi^*(X)\right]\right| \\
& && \text{(by Hoeffding's Inequality)} \\
=& \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + \left|\Pr\left[\phi_{k,n}^{\text{NN}}(X)=1\right] - \Pr\left[\phi^*(X)=1\right]\right| \\
=& \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + \left|\Pr\left[\phi_{k,n}^{\text{NN}}(X)\neq 0\right] - \Pr\left[\phi^*(X)\neq 0\right]\right| \\
\leq& \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + 2|R(\phi_{k,n}^{\text{NN}}) - R(\phi^*)| \\
\leq& \sqrt{\frac{1}{2n}\ln\frac{4}{\gamma}} + 12\sqrt{\frac{2c_d^2}{n}\ln\frac{4}{\gamma}}. && \text{(by Theorem 2.2)}
\end{aligned}$$

We note that the first equality follows the idea in the proof of Lemma B.2, by just replacing the Bayes classifier with the concrete k-NN classifier. □

Proof. [**Proof of Theorem 5.2**] To enhance the clarity of this proof, we will additionally assume that the curve $T^{(0)}$ is strictly decaying. We first need to understand the interpretation of lines 6 and 7 of the algorithm. The algorithm detects a violation, if

$$i^* > \tilde{\alpha}(\hat{\eta}^*) + w(\gamma),$$

where i^* solves the equation $T^{(0)}(i^*) = \tilde{\beta}(\hat{\eta}^*) + w(\gamma)$. We apply $T^{(0)}$ on both sides, which gives us the detection condition

$$\tilde{\beta}(\hat{\eta}^*) + w(\gamma) < T^{(0)}(\tilde{\alpha}(\hat{\eta}^*) + w(\gamma)). \tag{20}$$

Geometrically this means that the point $(\tilde{\alpha}(\hat{\eta}^*) + w(\gamma), \tilde{\beta}(\hat{\eta}^*) + w(\gamma))$ is below the curve $T^{(0)}$ and since $T^{(0)}$ is a trade-off curve, it follows that the entire box $\square_\gamma$ is below $T^{(0)}$. Conversely, if the detection condition is violated, we have

$$\tilde{\beta}(\hat{\eta}^*) + w(\gamma) \geq T^{(0)}(\tilde{\alpha}(\hat{\eta}^*) + w(\gamma)) \tag{21}$$

and the upper right edge point of the box $\square_\gamma$ is on or above $T^{(0)}$.

Now, suppose there was no violation (part 1) of the theorem). Then, any point on or above T is also on or above $T^{(0)}$. The point $(\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta))$ is on or above T and thus on or above $T^{(0)}$. Now, according to Theorem 5.1 it holds with probability $\geq 1-\gamma$ that the following event occurs

$$\mathcal{E} = \{(\mathbb{E}\tilde{\alpha}(\eta), \mathbb{E}\tilde{\beta}(\eta)) \in \square_\gamma\}$$

Conditional on that event the upper right edge point of $\square_\gamma$, namely $(\tilde{\alpha}(\hat{\eta}^*) + w(\gamma), \tilde{\beta}(\hat{\eta}^*) + w(\gamma))$ is also above T. It is hence above $T^{(0)}$ and satisfies condition (21) and no privacy violation is detected.

Now, in part 2), suppose that there exists a privacy violation. The trade-off function is strictly convex and it is not hard to see that this implies that it equals the set $\{(\alpha(\eta), \beta(\eta) : \eta \geq 0\}$ in this case (the constant λ in the Neyman-Pearson test can be set to 0 everywhere). We also define the maximum violation

$$v^* = \sup_{\alpha\in[0,1]}\left[T^{(0)}(\alpha) - T(\alpha)\right]$$

and the set of thresholds

$$\Psi := \left\{\eta \geq 0 : T^{(0)}(\alpha(\eta)) - T(\alpha(\eta)) \geq v^*/2\right\}.$$

It holds by the proof of Theorem 4.2 case 1) that

$$\sup_{\eta}|\hat{\alpha}_h(\eta) - \alpha(\eta)| \xrightarrow{P} 0, \quad as\ n_1 \to \infty.$$

In particular, it follows that

$$\Pr[\hat{\eta}^* \in \Psi] = 1 - r_{n_1},$$

where $r_{n_1} \to 0$ as $n_1 \to \infty$. If the above statement were false, it would follow on an event with asymptotically positive probability that

$$T^{(0)}(\alpha(\hat{\eta}^*)) - T(\alpha(\hat{\eta}^*)) \leq (1/2)v^*$$

leading to a contradiction with Proposition 4.3. Now, we condition on the event $\{\hat{\eta}^* \in \Psi\}$ and pass the parameter to the BayBox estimator, which returns the estimator pair $(\tilde{\alpha}(\hat{\eta}^*), \tilde{\beta}(\hat{\eta}^*))$. Now, keeping n_1 fixed and letting $n_2 \to \infty$ it follows that (part 2) of Theorem 5.1)

$$\tilde{\alpha}(\hat{\eta}^*) + w(\gamma) \xrightarrow{P} \alpha(\hat{\eta}^*), \quad \tilde{\beta}(\hat{\eta}^*) + w(\gamma) \to \beta(\hat{\eta}^*).$$

Given the continuity of the function $T^{(0)}$ (every trade-off function is continuous) it follows that conditionally on Ψ

$$\begin{aligned}
T^{(0)}(\tilde{\alpha}(\hat{\eta}^*) + w(\gamma)) \to T^{(0)}(\alpha(\hat{\eta}^*)) &\geq T(\alpha(\hat{\eta}^*)) + v^*/2 \\
=& \beta(\hat{\eta}^*) + \nu^*/2 > \beta(\hat{\eta}^*)
\end{aligned}$$

and the detection condition in (20) is asymptotically fulfilled as $n_2 \to \infty$. Thus, we have

$$\lim_{n_2 \to \infty} \Pr[A = \text{"Violation"} | \{\hat{\eta}^* \in \Psi\}] = 1$$

and hence

$$\liminf_{n_2 \to \infty} \Pr[A = \text{"Violation"}] \geq 1 - r_{n_1}.$$

Taking the limit $n_1 \to \infty$ we have $r_{n_1} \to 0$ and the result follows. □