Summary Statistic Privacy in Data Sharing

Zinan Lin, Shuaiqi Wang, *Graduate Student Member, IEEE*, Vyas Sekar, and Giulia Fanti, *Member, IEEE*

***Abstract*—We study a setting where a data holder wishes to share data with a receiver, *without* revealing certain summary statistics of the data distribution (e.g., mean, standard deviation). It achieves this by passing the data through a randomization mechanism. We propose *summary statistic privacy*, a metric for quantifying the privacy risk of such a mechanism based on the worst-case probability of an adversary guessing the distributional secret within some threshold. Defining distortion as a worst-case Wasserstein-1 distance between the real and released data, we prove lower bounds on the tradeoff between privacy and distortion. We then propose a class of quantization mechanisms that can be adapted to different data distributions. We show that the quantization mechanism's privacy-distortion tradeoff matches our lower bounds under certain regimes, up to small constant factors. Finally, we demonstrate on real-world datasets that the proposed quantization mechanisms achieve better privacy-distortion tradeoffs than alternative privacy mechanisms.**

***Index Terms*—Privacy, data privacy, synthetic data.**

I. Introduction

DATA sharing is an important enabler for data-driven product development [1], coordination efforts (e.g., cybersecurity [2], law enforcement [3]), and the creation of benchmarks for evaluating scientific progress [4], [5], [6]. However, *summary statistics* of shared data may leak sensitive information [7], [8]. For example, *property inference* attacks allow an attacker to infer properties about the individuals in the training dataset of a released machine learning model [9], [10], [11], [12], [13]. An institution that shares Domain Name System (DNS) data may not want to disclose even aggregated queries, as these quantities can be used to infer details about the institution [14]. A cloud provider that shares cluster performance traces may not want to reveal the proportions of different server types that the cloud provider owns, which are regarded as business secrets [15]. Note that this information (aggregate DNS queries, proportions of server types), cannot be inferred from any record, but is a property of the data distribution (or the aggregate dataset). We therefore define these secrets in Section III as functions that can be computed from one or more parameters of the data distribution.

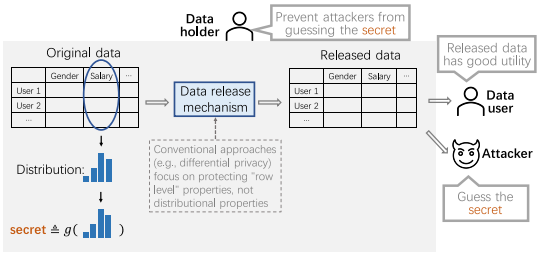


Fig. 1. Problem overview. The data holder wants to release data while hiding *statistical secrets* of the original data. The attacker (which could be the data user) observes the released data, and wants to guess the *secrets* of the original data. We focus on secrets defined over the underlying distribution (e.g., functions of the moments or quantiles of data *columns*). Many existing frameworks (e.g., differential privacy [16]) protect information from *samples (rows)*.

Our setup is as follows (detailed formulation in Section III). A data holder possesses a data distribution. The data holder chooses one or more secrets, which are defined as deterministic functions of the distribution. For example, a video analytics company might choose the mean daily observed traffic as a secret quantity. Then, the data holder obfuscates their data distribution according to a randomization *mechanism* and releases the output (Fig. 1). The goal is to prevent an adversary from estimating the value of the secrets, while preserving data utility.

Many widely-used privacy metrics and data sharing algorithms are not designed to protect summary statistic privacy, instead protecting the privacy of individual records in a database (e.g., differential privacy [16], anonymization [17], sub-sampling [17]). For example, differential privacy (DP) [16] evaluates how much individual samples influence the final output of an algorithm, and does not inherently protect summary statistics [9].

Many other frameworks have been designed specifically to hide aggregate properties of a dataset (or a distribution) [18], [19], [20]; we discuss these in detail in Section II. Many of these frameworks define privacy in terms of information-theoretic quantities such as mutual information [19] or other divergences [21]. In this work, we directly define the *privacy* of a mechanism as the posterior probability that a worst-case attacker can infer the data holder's true secret after observing the released data. This definition is related to prior work analyzing min-entropy as a privacy metric [22]. To capture the utility of released data, we define the *distortion* of a mechanism as the worst-case Wasserstein-1 distance between the original and released data distributions, given that data release typically occurs in

Manuscript received 27 October 2023; revised 14 January 2024 and 16 April 2024; accepted 15 May 2024. Date of publication 21 May 2024; date of current version 17 June 2024. This work was supported in part by NSF under Grant CIF-1705007 and Grant RINGS-2148359; in part by the Sloan Foundation; in part by Intel; in part by J.P. Morgan Chase; in part by Siemens; in part by Bosch; in part by Cisco; and in part by the U.S. Army Research Office and the U.S. Army Futures Command under Contract W911NF20D0002. *(Zinan Lin and Shuaiqi Wang contributed equally to this work.) (Corresponding author: Shuaiqi Wang.)*

Zinan Lin is with the Algorithms Group, Microsoft Research, Redmond, WA 98052 USA (e-mail: zinanlin@microsoft.com).

Shuaiqi Wang, Vyas Sekar, and Giulia Fanti are with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA 15213 USA (e-mail: shuaiqiw@andrew.cmu.edu; vsekar@andrew.cmu.edu; gfanti@andrew.cmu.edu).

This article has supplementary downloadable material available at https://doi.org/10.1109/JSAIT.2024.3403811, provided by the authors.

Digital Object Identifier 10.1109/JSAIT.2024.3403811

one shot. Our goal is to design data release mechanisms that efficiently trade off privacy and distortion (defined in Section III).

A. Contributions

Our contributions are as follows.

- *Lower bounds (Section IV):* We derive general lower bounds on distortion given a privacy budget for any mechanism. These bounds depend on both the secret function and the data distribution. We derive closed-form lower bounds for a number of case studies (i.e., combinations of prior beliefs on the data distribution and secret functions).
- *Mechanism design and upper bounds (Section V):* We propose a class of mechanisms that achieve summary statistic privacy called *quantization mechanisms*, which intuitively quantize a data distribution's parameters[1] into bins. We show that for the case studies analyzed theoretically in Table I, the quantization mechanism achieves a privacy-distortion tradeoff within a small constant factor of optimal (usually $\leq$3) in the regime where quantization bins are small relative to the overall support set of the distribution parameters. We present a *sawtooth technique* for theoretically analyzing the quantization mechanism's privacy tradeoff under various types of secret functions and data distributions (Section V-C). Intuitively, the sawtooth technique exploits the geometry of the distribution parameter(s) to divide the parametric space into two regions: one in which privacy risk is small and analytically tractable, and another in which privacy risk can be high, but which occurs with low probability. For the case studies that we do not analyze theoretically, we provide a dynamic programming algorithm that efficiently numerically instantiates the quantization mechanism.
- *Empirical evaluation (Section VII):* We give empirical results showing how to use summary statistic privacy to release a real dataset, and how to evaluate the corresponding summary statistic privacy metric. We show that the proposed quantization mechanism achieves better privacy-distortion tradeoffs than other related privacy mechanisms.

II. Related Work

We divide the related work into two categories: approaches based on indistinguishability over candidate inputs, and information-theoretic approaches.

A. Indistinguishability-Based Approaches

Differential privacy (DP) [16] is one of the most commonly-adopted privacy frameworks. A random mechanism $\mathcal{M}$ is (ϵ, δ)-differentially-private if for any neighboring datasets $\mathcal{X}_0$ and $\mathcal{X}_1$ (i.e., $\mathcal{X}_0$ and $\mathcal{X}_1$ differ one sample), and any set $S \subseteq range(\mathcal{M})$, we have

$$\mathbb{P}(\mathcal{M}(\mathcal{X}_0) \in S) \leq e^{\epsilon} \cdot \mathbb{P}(\mathcal{M}(\mathcal{X}_1) \in S) + \delta.$$

[1]We assume data distributions are drawn from a parametric family; more details in Section III.

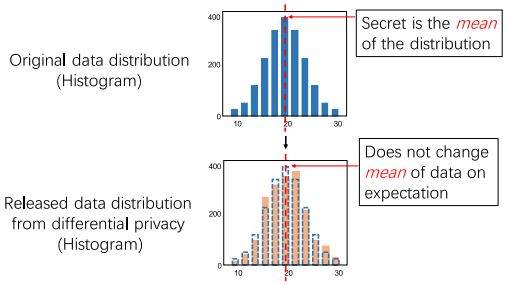


Fig. 2. An illustrative example of why naive differential privacy mechanisms do not protect summary statistics. Suppose we want to protect the *mean* of the data. A typical differential privacy algorithm [23] would add zero-mean noise (e.g., Laplace noise) to the bins. This mechanism does not change the expected mean of the data.

A natural first attempt at the problem might use DP by treating $\mathcal{M}$ as the data release mechanism that takes the original dataset as input and outputs the released dataset. For example, suppose we want to release the histogram in Fig. 2, representing the number of items sold by a company at different prices. Suppose the mean of this distribution is sensitive, as it can be used to determine the company's overall trade volume. Two natural approaches for using DP arise:

(1) *Per-record DP:* A typical DP algorithm [23] would add zero-mean noise (e.g., Laplace noise) to each histogram bin in Fig. 2. This prevents the adversary from inferring whether any individual record was in the dataset, but allows the attacker to derive an unbiased estimator of the mean from the released data. In other words, the threat model of (this usage of) DP and our framework are different: DP hides whether *any given sample* contributed to the shared data, whereas we want to hide functions of *the underlying distribution*. To show this formally, one can construct counterexamples where a data release mechanism is DP, but cannot protect a summary statistic of a data distribution (or dataset). We construct such a counterexample in Appendix A-A in the supplementary material for the scenario of hiding the mean of a dataset of scalar numbers. The example shows that if the data holder applies a local DP mechanism [24] (Gaussian mechanism) to their dataset, as the number of dataset samples n grows, the released noisy mean concentrates around the original mean. Hence, the adversary can guess the mean to within a tolerance $\epsilon > 0$ with a probability that tends to 1 as $n \to \infty$.

(2) *Per-attribute-per-record DP:* One can also design a local DP mechanism that adds independent noise to each record of the dataset, with independent noise of different scales used for different attributes. Consider an example of hiding the difference in mean salaries between males and females in a gender-salary dataset. A data holder could use a local DP mechanism that adds independent noise of different scales to each of the gender and salary attributes. Since the mechanism itself is locally DP, it provides privacy guarantees at the level of each record, as well as each attribute of each record (i.e., the DP privacy guarantee protects individual cells in the dataset). However, such a class of mechanism still cannot hide distributional properties (in our example, the mean salary difference between males and females). In Appendix A-B in the supplementary material, we precisely formulate and

analyze this example, and show that there exists an attack strategy such that the probability the adversary guesses the secret to within tolerance $\epsilon > 0$ tends to 1 as dataset size $n \to \infty$.

(3) *Per-dataset DP:* A third natural alternative is to devise a DP-like definition that explicitly protects the secret quantity. For instance, we could ask that for any pair of input distributions that differ in their secret quantity, the data release mechanism outputs similar released data distributions. Several per-dataset methods are listed below. Used naively, such an approach provides strong privacy guarantees, but may have poor utility. For instance, consider two Gaussian input distributions $\mathcal{N}(\mu_1, \sigma_1^2)$ and $\mathcal{N}(\mu_2, \sigma_2^2)$ with the secret as the mean. The values of σ_1^2 and σ_2^2 could be arbitrarily different. To make input distributions indistinguishable given the released data, we must destroy information about the true σ, which requires adding potentially unbounded noise. While relaxations like metric differential privacy may help [25], they may introduce new challenges, e.g., how to choose the metric function to map dataset distance to a privacy parameter.

Attribute privacy [18] tackles these challenges in part by constraining the space of distributions that should be indistinguishable [11]. Attribute privacy protects a function of a sensitive column in the dataset (named *dataset attribute privacy*) or a sensitive parameter of the underlying distribution from which the data is sampled (named *distribution attribute privacy*). It addresses the previously-mentioned shortcomings of vanilla DP under the *pufferfish privacy framework* [26]. Precisely, let $\mathcal{X}$ be the dataset, $\mathcal{G}$ be the possible range of a secret g, and $\mathcal{G}_a, \mathcal{G}_b \subseteq \mathcal{G}$ be two non-overlapping subsets of the secret range $\mathcal{G}$. A mechanism $\mathcal{M}$ is (ϵ, δ)-attribute private if for any dataset $\mathcal{X}$, secret range pairs $\mathcal{G}_a, \mathcal{G}_b$, and any set $S \subseteq range(\mathcal{M})$:

$$\mathbb{P}(\mathcal{M}(\mathcal{X}) \in S | g(\mathcal{X}) \in \mathcal{G}_a) \leq e^{\epsilon} \mathbb{P}(\mathcal{M}(\mathcal{X}) \in S | g(\mathcal{X}) \in \mathcal{G}_b) + \delta.$$

Attribute privacy focuses on algorithms that output *a statistical query of the dataset* instead of the entire dataset. Though we may apply attribute privacy to analyze full-dataset-sharing algorithms; it may need to add substantial noise due to the high dimensionality of the dataset (Section VII).

Distribution privacy [27] is a closely related notion, which releases a full data distribution under DP-style indistinguishability guarantees. Roughly, for any two input distributions with parameters θ_0 and θ_1 from a pre-defined set of candidate distributions, a distribution-private mechanism outputs a distribution $\mathcal{M}(\theta_i)$ for $i \in \{0, 1\}$ such that for any set S in the output space, we have $\mathbb{P}[\mathcal{M}(\theta_i) \in S] \leq e^{\epsilon} \mathbb{P}[\mathcal{M}(\theta_{1-i}) \in S] + \delta$. By obfuscating the whole distribution, distribution privacy inherently protects the private information. However the required noise may be more than what is needed to protect only select secret(s). For example, as mentioned above, two datasets can have *exactly the same* secret statistic (e.g., mean), while differing significantly in other respects (e.g., variance)—this requires significant noise in general. A recent work [28] proposes mechanisms for distribution privacy, and we observe this trend experimentally in Section VII; the noise added by the mechanisms in [28] is larger than what we require with summary statistic privacy (though the privacy guarantees are different, so it is difficult to do a fair comparison).

Distribution inference [7], [8] considers a hypothesis test in which the adversary must choose whether released data comes from one of two fixed input data distributions ω_1, ω_2. Both distributions are assumed to be known to all parties. By defining the attacker's guessed distribution as $\hat{\omega}$ and the attacker's advantage as $|\mathbb{P}(\hat{\omega}|\omega_1) - \mathbb{P}(\hat{\omega}|\omega_2)|$, distribution inference requires that the attacker's advantage be negligible. However, it is unclear how to establish a reasonable pair of candidate distributions; moreover, as with distribution privacy and attribute privacy, distribution inference may require high noise since it requires the data distributions to be indistinguishable.

B. Information-Theoretic Approaches

The second category of frameworks use information-theoretic measures of privacy and utility [20], [22], [29], [30], [31], [32], [33], [34], [35], [36], [37], [38], [39], [40], [41], [42], [43], [44]. Such works often measure disclosure via divergences, such as mutual information [19], [34], [45], [46], [47], [48], [49], [50], [51], f-divergences [21], [52], or min-entropy [22], [30], [31], [38], [39], [40], [53], [54], [55]. We discuss a few examples here.

Privacy funnel [19], [29] is a well-known information-theoretic privacy framework. Let X be the random variable of the original data, containing sensitive information U, and let Y represent the (random) released data. The privacy funnel framework evaluates privacy leakage with the mutual information $I(U; Y)$, and the utility of Y with mutual information $I(X; Y)$. To find a data release mechanism $P_{Y|X}$, privacy funnel solves the optimization $\min_{P_{Y|X}: I(X;Y) \geq R} I(U; Y)$, where R is a desired threshold on the utility of Y. Adopting the same privacy and utility metrics, Zamani et al. [46] instead analyze an upper bound on utility under a privacy constraint, i.e., $\sup_{P_{Y|X}: I(U;Y) \leq \epsilon} I(X; Y)$, where ϵ represents the privacy constraint. However, prior work has argued that mutual information is not a good metric for either privacy or utility [20]. On the privacy front, there exist mechanisms that reduce $I(U; Y)$ while allowing the attacker to guess U correctly from Y with higher probability (see [20, Example 1]). On the utility front, high mutual information $I(X; Y)$ does not mean that the released data Y is a useful representation of X; for instance, Y could be an arbitrary one-to-one transformation of X.

Rate-distortion formulations. Although rate-distortion theory was originally proposed in the context of source coding [56], it has more recently been used to model privacy problems as follows. Let X be the random variable with finite or countable alphabets $\mathbb{X}$ with prior distribution π, and $\mathcal{M}$ be the mechanism that encodes $\mathbb{X}$ to $\mathbb{Y}$. For a distortion $d : \mathbb{X} \times \mathbb{Y} \to \mathbb{R}_{\geq 0}$ and a threshold $\hat{d}$, the problem is to find the optimal mechanism that minimizes MI subject to distortion constraint $\hat{d}$: $\mathcal{M}^* = \arg\min_{\mathbb{E}[D(X,\mathcal{M},d)] \leq \hat{d}} I(X; Y_{X,\mathcal{M}})$, where $Y_{X,\mathcal{M}}$ is the encoding of X. Several papers have used this formulation to model tradeoffs between privacy (mutual information) and utility (distortion), particularly in the context of location

privacy [47], [48]. These works use the celebrated Blahut-Arimoto algorithm [57], [58] to identify a mechanism that Pareto-optimally trades off mutual information for average distortion.

As mentioned earlier, mutual information has some shortcomings as a privacy metric [20]. Nevertheless, the rate-distortion formulation of privacy is related to our work in that it uses distortion as the measure of utility. Whereas rate-distortion-based formulations use average-case distortion as a distortion metric, we use worst-case distortion (Section III). If we replace the average distortion in rate-distortion theory by the worst-case distortion, finding the Pareto-optimal mechanism may be substantially more challenging since the objective function $\mathcal{M}^* = \arg\min_{\max_{x\in X, y\in Y_{x,\mathcal{M}}} d(x,y)\le \hat{d}} I(X; Y_{X,\mathcal{M}})$ is no longer a convex program in general.

Maximal leakage [20] is an information-theoretic framework for quantifying the leakage of sensitive information. Using the same notation as before, the adversary's guess of secret U is denoted by $\hat{U}$. Based on this setup, the Markov chain $U - X - Y - \hat{U}$ holds. Maximal leakage $\mathcal{L}$ from X to Y is defined as

$$\mathcal{L}(X \to Y) = \sup_{U-X-Y-\hat{U}} \log \frac{\mathbb{P}\left(U = \hat{U}\right)}{\max_u P_U(u)}, \tag{1}$$

where the sup is taken over U (i.e., considering the worst-case secret) and $\hat{U}$ (i.e., considering the strongest attacker). Intuitively, Eq. (1) evaluates the ratio (in nats) of the probabilities of guessing the secret U correctly with and without observing Y. Variants and generalizations of maximal leakage have been proposed, modifying Eq. (1) to penalize different values of $\mathbb{P}\left(U = \hat{U}\right)$ differently, using so-called *gain functions* [33], [35], [36], [37]. Maximal leakage and its variants assume that the secret U is unknown a priori and therefore considers the worst-case leakage over all possible secrets. However, in our problem, data holders know what secret they want to protect.

Min-entropy metrics: Several papers have studied privacy metrics related to min-entropy, or the probability of guessing the secret correctly [22], [30], [31], [53], [54]. Among these, the most closely related paper is by Asoodeh et al. [22], which directly analyzes the probability of guessing the secret, as we do (within a threshold). Adopting the same notation as before (i.e., the Markov chain $U - X - Y$), [22] aims to maximize the disclosure of X (i.e., $\max_f \mathbb{P}(X = f(Y))$, where the max is taken over all functions f) to ensure high utility. This optimization is subject to a privacy constraint on the sensitive information U: $\max_{\hat{g}} \mathbb{P}\left(U = \hat{g}(Y)\right) \le T$, where the max is taken over all attack strategies $\hat{g}$. However, the authors assume that for random variables X and Y, the value of each dimension can only be either 0 or 1 (i.e., each dimension of the data distribution parameter is binary). Since their analysis relies on the properties of Bernoulli distribution, the results cannot be trivially extended to non-binary case, significantly constraining the range of distribution settings this framework can analyze. Furthermore, they assess utility based on the probability of precisely guessing the original data. However, in data-sharing contexts, this utility measure suffers from the same shortcomings as mutual information, namely that any random one-to-one mapping can achieve a high utility metric without having practical utility.

Quantitative information flow: The concept of quantitative information flow (QIF) was first introduced in [59], [60], with the goal of quantitatively measuring the amount of information leaked about a secret by observing the output data. QIF broadly encompasses several of the privacy frameworks we have mentioned previously. Early works mainly adopted mutual information as the leakage definition [49], [50], [51], while Smith [30] showed that it fails to capture *vulnerability*: the probability of an attacker successfully guessing the secret in one try. This led to the introduction of min-entropy leakage, which is a normalized variant of our privacy metric. Generalizations of min-entropy leakage have been proposed [31], [53], [55]; in particular, g-leakage [53] introduces a gain function that models partial guessing or multiple guessing scenarios. Recently, g-leakage was used as a privacy metric to study a variety of applications, including the combination of local DP and shuffling [38], average-case utility in privacy-preserving pipelines [40], and cyber-attack defense problems [39]. However, unlike our work, these works consider the entire input dataset as the secret information that needs to be protected, whereas we only need to protect the sensitive information U contained in X, while maximizing the disclosure of nonsensitive information in X. Because of this, our goal is to minimize the information leakage of U while maximizing the leakage of other information in X, and to derive fundamental limits on tradeoffs between these quantities. Although we could have used min-entropy leakage in our privacy metric design, their analysis applies to discrete alphabets and cannot be trivially extended to the continuous case, as the probability density of an attacker guessing the exact secret is zero. Moreover, the only utility analysis in these works is average-case, whereas our paper adopts a worst-case utility metric, which significantly changes the mechanisms and conclusions we draw (e.g., leading to quantization-based mechanisms).

Noiseless privacy-preserving policies: An interesting property of the mechanism we study—the quantization mechanism—is that it is deterministic. Several prior works have studied noiseless privacy mechanisms under various assumptions on the generative process for the data [41], [42], [43], [44]. For example, adopting non-stochastic information theoretic methods, Farokhi [42] use *maximin information* [61] and *non-stochastic information leakage* as privacy metrics and measure utility as the worst-case difference between the input and output responses. For instance, *maximin information* is defined as $I(\mathcal{X}; \mathcal{Y}) = \log(|\Upsilon(\mathcal{X}, \mathcal{Y})|)$; here $\mathcal{X}, \mathcal{Y}$ represent the original and released datasets, and $\Upsilon(\mathcal{X}, \mathcal{Y})$ represents the unique taxicab partition of $[\![\mathcal{X}, \mathcal{Y}]\!]$, where $[\![\mathcal{X}, \mathcal{Y}]\!]$ denotes the set of all feasible input-output dataset pairs. Roughly, a taxicab partition consists of a sequence of dataset pairs such that any two consecutive pairs share the same $\mathcal{X}$ or $\mathcal{Y}$ (formal definition in [42]). Reference [42] proves that the quantization mechanism is the optimal privacy-preserving policy over the set of deterministic piecewise differentiable mechanisms (which quantizes the input $\mathcal{X}$ as several bins and

the output $\mathcal{Y}$ is differentiable for each bin) that maximize the privacy level subject to a utility constraint for maximin information.

In our work, despite not constraining the mechanism space to deterministic mechanisms, we also find that the quantization mechanism is near-optimal. However, the proof in [42] is based on the property of taxicab connectivity [61] from non-stochastic information theory, which cannot be directly adopted in our analysis. The similarities in our findings (albeit over quite different problem formulations and analysis techniques) suggest that quantization-based mechanisms may be a universally good solution for private data release problems subject to worst-case distortion constraints. This question may be an interesting direction for future work.

III. Problem Formulation

Notation: We denote random variables with uppercase English letters or upright Greek letters (e.g., X, μ), and their realizations with italicized lowercase letters (e.g., x, μ). For a random variable X, we denote its probability density function (PDF) as f_X, and its distribution measure as ω_X. If a random variable X is drawn from a parametric family (e.g., Gaussian with specified mean and covariance), the parameters will be denoted with a subscript of X, i.e., the above notations become X_θ, f_{X_θ}, ω_{X_θ} respectively for parameters $\theta \in \mathbb{R}^q$, where $q \geq 1$ denotes the dimension of the parameters. In addition, we denote $f_{X|Y}$ as the conditional PDF or PMF of X given another random variable Y. We use $\mathbb{Z}, \mathbb{Z}_{>0}, \mathbb{N}, \mathbb{R}, \mathbb{R}_{>0}$, to denote the set of integers, positive integers, natural numbers, real numbers, and positive real numbers, respectively.

Original data: Consider a data holder who possesses a dataset of n samples $\mathcal{X} = \{x_1, \ldots, x_n\}$, where for each $i \in [n]$, $x_i \in \mathbb{R}$ is drawn i.i.d. from an underlying distribution. We assume the distribution comes from a parametric family, and the parameter vector $\theta \in \mathbb{R}^q$ of the distribution fully specifies the distribution. That is, $x_i \sim \omega_{X_\theta}$, where we further assume that θ is itself a realization of random parameter vector Θ, and ω_Θ is the probability measure for Θ. We will discuss how to relax the assumption on this prior distribution of θ in Section VIII. We assume that the data holder knows θ (and hence knows its full data distribution ω_{X_θ}); our results and mechanisms generalize to the case when the data holder only possesses the dataset $\mathcal{X}$ (see Section VI).

For example, suppose the original data samples come from a Gaussian distribution. We have $\theta = (\mu, \sigma)$, and $X_\theta \sim \mathcal{N}(\mu, \sigma)$. ω_Θ (or f_Θ) describes the prior distribution over (μ, σ). For example, if we know a priori that the mean of the Gaussian is drawn from a uniform distribution between 0 and 1, and σ is always 1, we could have $f_\Theta(\mu, \sigma) = \mathbb{I}(\mu \in [0, 1]) \cdot \delta(\sigma)$, where $\mathbb{I}(\cdot)$ is the indicator function, and δ is the Dirac delta function.

Statistical secret to protect: We assume the data holder wants to hide a secret quantity, which is defined as a function of the original data distribution. Since the true data distribution is fully specified by parameter vector θ, we define the secret as a function of θ as follows: $g(\theta)$:$\mathbb{R}^q \to \mathbb{R}$. In the Gaussian example $X_\theta \sim \mathcal{N}(\mu, \sigma)$, suppose the data holder wishes to hide the mean; we thus have that $g(\mu, \sigma) = \mu$.

Data release mechanism: The data holder releases data by passing the private parameter θ through a *data release mechanism* $\mathcal{M}_g$. That is, for a given θ, the data holder first draws internal randomness $z \sim \omega_Z$, and then releases another distribution parameter $\theta' = \mathcal{M}_g(\theta, z)$, where $\mathcal{M}_g$ is a deterministic function, and ω_Z is a fixed distribution from which z is sampled. Note that we assume both the input and output of $\mathcal{M}_g$ are distribution parameters. It is straightforward to generalize to the case when the input and/or output are datasets of samples (see Section VI).

For example, in the Gaussian case discussed above, one data release mechanism could be $\mathcal{M}_g((\mu, \sigma), z) = (\mu + z, \sigma)$ where $z \sim \mathcal{N}(0, 1)$. I.e., the mechanism shifts the mean by a random amount drawn from a standard Gaussian distribution and keeps the variance.

Threat model: We assume that the attacker knows the parametric family from which the data is drawn, and has a prior over the parameter realization, but does not know the initial parameter θ. The attacker also knows the data release mechanism $\mathcal{M}_g$ and output θ' but not the realization of the data holder's internal randomness z. The attacker guesses the initial secret $g(\theta)$ based on the released parameter θ' according to estimate $\hat{g}(\theta')$. $\hat{g}$ can be either random or deterministic, and we assume no computational bounds on the adversary. For instance, in the running Gaussian example, an attacker may choose $\hat{g}(\mu', \sigma') = \mu'$.

Privacy metric: The data holder wishes to prevent an attacker from guessing its secret $g(\theta)$.

Inspired by min-entropy, we define our privacy metric privacy $\Pi_{\epsilon,\omega_\Theta}$ as the attacker's probability of guessing the secret(s) to within a tolerance ϵ, taken worst-case over all attackers $\hat{g}$:

$$\Pi_{\epsilon,\omega_\Theta} \triangleq \sup_{\hat{g}} \mathbb{P}\big(|\hat{g}(\theta') - g(\theta)| \leq \epsilon\big). \tag{2}$$

The probability is taken over the randomness of the original data distribution ($\theta \sim \omega_\Theta$), the data release mechanism ($z \sim \omega_Z$), and the attacker strategy ($\hat{g}$).

Remark: This privacy metric is an average-case guarantee over the prior distribution of the parameters ω_Θ. A natural question is whether this can be converted into a worst-case privacy guarantee, as is common in many privacy frameworks [16], [18], [28]. However, a worst-case variant of the metric (worst-case over prior distributions) is too stringent; no mechanism can achieve meaningful privacy (< 1), as formally stated below (proof in Appendix D-A in the supplementary material).

Proposition 1: There is no data release mechanism whose privacy value satisfies

$$\sup_{\omega_\Theta} \Pi_{\epsilon,\omega_\Theta} < 1.$$

Distortion metric: The main goal of data sharing is to provide useful data; hence, we (and data holders and users) want to understand how much the released data distorts the original

data. We define the *distortion* Δ of a mechanism as the worst-case distance between the original distribution and the released distribution:

$$\Delta \triangleq \sup_{\substack{\theta\in\text{Supp}(\omega_\Theta),\theta',\\ z\in\text{Supp}(\omega_Z):\mathcal{M}_g(\theta,z)=\theta'}} d\big(\omega_{X_\theta}\|\omega_{X_{\theta'}}\big), \tag{3}$$

where d is Wasserstein-1 distance. We use a worst-case definition of distortion because in data sharing settings, data is typically released in one shot, so that data should be useful even in the worst case. Wasserstein-1 distance is commonly used as the distance metric in neural network design (e.g., [62], [63]). Note that the definition in Eq. (3) can be extended to data release mechanisms that take datasets as inputs and/or outputs.

Formulation: To summarize, the data holder's objective is to choose a data release mechanism that minimizes distortion Δ subject to a constraint on privacy $\Pi_{\epsilon,\omega_\Theta}$:

$$\begin{aligned}&\min_{\mathcal{M}_g} \ \Delta\\ &\text{subject to} \ \ \Pi_{\epsilon,\omega_\Theta} \le T.\end{aligned} \tag{4}$$

The reverse formulation, $\min_{\mathcal{M}_g}\Pi_{\epsilon,\omega_\Theta}$ subject to $\Delta \le T$ is analyzed in Appendix B in the supplementary material.

The optimal data release mechanisms for Eq. (4) depends on the secrets and the characteristics of the original data. Data holders specify the secret function they want to protect and select the data release mechanism to process the raw data for sharing.

Our goal is to study: (1) What are fundamental limits on the tradeoff between privacy and distortion? (2) Do there exist data release mechanisms that can match or approach these fundamental limits? In general, these questions can have different answers for different parametric families of data distributions and secret functions. In Sections IV and V, we first present general results that do not depend on data distribution or secret function. We then present case studies for specific secret functions and data distributions in Section VI.

IV. General Lower Bound on Privacy-Distortion Tradeoffs

Given a privacy budget T, we first present a lower bound on distortion that applies regardless of the prior distribution of data ω_Θ and regardless of the secret g. In other words, this applies for arbitrary correlations between parameters, which are captured by the prior ω_Θ.

Theorem 1 (Lower Bound of Privacy-Distortion Tradeoff): Let $D\big(X_{\theta_1},X_{\theta_2}\big) \triangleq \frac{1}{2}d\Big(\omega_{X_{\theta_1}}\|\omega_{X_{\theta_2}}\Big)$, where $d(\cdot\|\cdot)$ denotes Wasserstein-1 distance. Further, let $R\big(X_{\theta_1},X_{\theta_2}\big) \triangleq |g(\theta_1)-g(\theta_2)|$ and

$$\gamma \triangleq \inf_{\theta_1,\theta_2\in\text{Supp}(\omega_\Theta)} \frac{D\big(X_{\theta_1},X_{\theta_2}\big)}{R\big(X_{\theta_1},X_{\theta_2}\big)}. \tag{5}$$

For any $T\in(0,1)$, when $\Pi_{\epsilon,\omega_\Theta}\le T$,

$$\Delta > \left(\lceil\frac{1}{T}\rceil - 1\right)\cdot 2\gamma\epsilon. \tag{6}$$

The proof is shown as below. From Theorem 1 we see that the lower bound of distortion scales inversely with the privacy budget and positively with the tolerance threshold ϵ. The dependent quantity γ in Eq. (5) can be thought of as a conversion factor that bounds the translation from probability of detection to distributional distance. Note that we have not made γ exact as its form depends on the type of the secret and prior distribution of data. We will instantiate it in the cases studies in Section VI.

Proof: Our proof proceeds by constructing an ensemble of attackers, such that at least one of them will be correct by construction. We do this by partitioning the space of possible secret values, and having each attacker output the midpoint of one of the subsets of the partition. We then use the fact that each attacker can be correct with probability at most T, combined with γ, which intuitively relates the distance between distributions to the distance between their secrets, to derive the claim. Recall that θ is the true private parameter vector, θ' is the released parameter vector as a result of the data release mechanism.

$$\begin{aligned}T &\ge \Pi_{\epsilon,\omega_\Theta}\\ &= \sup_{\hat{g}}\mathbb{P}\big(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\big)\\ &= \sup_{\hat{g}}\mathbb{E}\left(\mathbb{P}\left(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\middle|\theta'\right)\right)\\ &= \mathbb{E}\left(\sup_{\hat{g}}\mathbb{P}\left(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\middle|\theta'\right)\right),\end{aligned} \tag{7}$$

where Eq. (7) is due to the following facts:

- LHS $\le$ RHS, as $\sup_{\hat{g}}\mathbb{P}\big(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]|\theta'\big) \ge \mathbb{P}\big(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]|\theta'\big)$ for any θ'.
- RHS $\le$ LHS: Let us define

$$t_{\theta'} \triangleq \sup_{\hat{g}}\mathbb{P}\left(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\middle|\theta'\right).$$

RHS$=\int_{\theta'}f_{\Theta'}(\theta')t_{\theta'}d\theta'$. We can define an attacker as $\hat{g}(\theta')=t_{\theta'}$. In that case,

$$\begin{aligned}&\mathbb{E}\left(\mathbb{P}\left(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\middle|\theta'\right)\right)\\ &\quad=\int_{\theta'}f_{\Theta'}(\theta')t_{\theta'}d\theta'.\end{aligned}$$

Therefore, LHS$\ge$RHS.

Thus, there exists θ' s.t.

$$\sup_{\hat{g}}\mathbb{P}\left(\hat{g}(\theta')\in\big[g(\theta)-\epsilon,g(\theta)+\epsilon\big]\middle|\theta'\right)\le T.$$

Let

$$\begin{aligned}L_{\theta'} &\triangleq \inf_{\theta\in\text{Supp}(\omega_\Theta),z:\mathcal{M}_g(\theta,z)=\theta'} g(\theta),\\ R_{\theta'} &\triangleq \sup_{\theta\in\text{Supp}(\omega_\Theta),z:\mathcal{M}_g(\theta,z)=\theta'} g(\theta).\end{aligned}$$

We can define a sequence of attackers and a constant N such that $\hat{g}_i(\theta')=L_{\theta'}+(i+0.5)\cdot 2\epsilon$ for $i\in\{0,1,\ldots,N-1\}$ and $L_{\theta'}+2N\epsilon\ge R_{\theta'}>L_{\theta'}+2(N-1)\epsilon$ (Fig. 3). From the above,

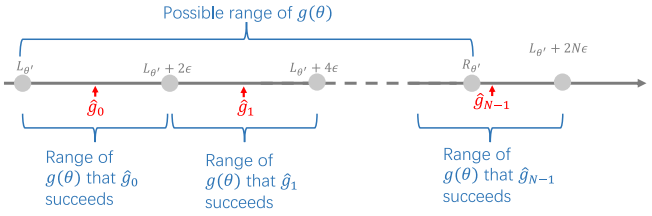


Fig. 3. The construction of attackers for proof of Theorem 1. The 2ϵ ranges of $\hat{g}_0, \ldots, \hat{g}_{N-1}$ jointly cover the entire range of possible secret $[L_{\theta'}, R_{\theta'}]$. The probability of guessing the secret correctly for any attacker is $\leq T$. Therefore, $R_{\theta'} - L_{\theta'} > \left(\lceil \frac{1}{T} \rceil - 1\right) \cdot 2\epsilon$ (Eq. (8)).

we have

$$T \cdot N \geq \sum_i \mathbb{P}\left(\hat{g}_i(\theta') \in \left[g(\theta) - \epsilon, g(\theta) + \epsilon\right] \middle| \theta'\right) \geq 1,$$

Therefore, we have $N \geq \lceil \frac{1}{T} \rceil$, and

$$R_{\theta'} - L_{\theta'} > \left(\lceil \frac{1}{T} \rceil - 1\right) \cdot 2\epsilon. \tag{8}$$

Then we have

$$\begin{aligned}\Delta &\geq \sup_{\theta \in \mathrm{Supp}(\omega_\Theta), z \in \mathrm{Supp}(\omega_Z): \mathcal{M}_g(\theta, z) = \theta'} d\left(\omega_{X_\theta} \| \omega_{X_{\theta'}}\right) \\ &\geq \sup_{\theta_i \in \mathrm{Supp}(\omega_\Theta), z_i: \mathcal{M}_g(\theta_i, z_i) = \theta'} D\left(X_{\theta_1}, X_{\theta_2}\right) \qquad (9) \\ &> \left(\lceil \frac{1}{T} \rceil - 1\right) \cdot 2\gamma\epsilon. \qquad (10)\end{aligned}$$

where in Eq. (9), θ_i for $i \in \{1, 2\}$ denotes two arbitrary parameter vectors in the support space, and Eq. (9), Eq. (10) are derived as follows:

- Eq. (9): Let

  $$\theta_1', \theta_2' = \arg \sup_{\theta_i \in \mathrm{Supp}(\omega_\Theta): \exists z_i s.t. \mathcal{M}_g(\theta_i, z_i) = \theta'} D\left(X_{\theta_1}, X_{\theta_2}\right).$$

  From the triangle inequality, we know that $d\left(\omega_{X_{\theta_1'}} \| \omega_{X_{\theta_2'}}\right) \leq d\left(\omega_{X_{\theta_1'}} \| \omega_{X_{\theta'}}\right) + d\left(\omega_{X_{\theta_2'}} \| \omega_{X_{\theta'}}\right)$. By definition, we also know that for $i = 1, 2$,

  $$d\left(\omega_{X_{\theta_i'}} \| \omega_{X_{\theta'}}\right) \leq \sup_{\theta \in \mathrm{Supp}(\omega_\Theta), z \in \mathrm{Supp}(\omega_Z): \mathcal{M}_g(\theta, z) = \theta'} d\left(\omega_{X_\theta} \| \omega_{X_{\theta'}}\right).$$

  Therefore, we have

  $$\mathrm{RHS} \leq \frac{1}{2} d\left(\omega_{X_{\theta_1'}} \| \omega_{X_{\theta'}}\right) + \frac{1}{2} d\left(\omega_{X_{\theta_2'}} \| \omega_{X_{\theta'}}\right) = \mathrm{LHS}.$$

- Eq. (10): Let

  $$\theta_1', \theta_2' = \arg \sup_{\theta_i \in \mathrm{Supp}(\omega_\Theta): \exists z_i s.t. \mathcal{M}_g(\theta_i, z_i) = \theta'} D\left(X_{\theta_1}, X_{\theta_2}\right),$$
  $$\theta_1'', \theta_2'' = \arg \inf_{\theta_i \in \mathrm{Supp}(\omega_\Theta): \exists z_i s.t. \mathcal{M}_g(\theta_i, z_i) = \theta'} R\left(X_{\theta_1}, X_{\theta_2}\right).$$

  We have

  $$\begin{aligned}\gamma &\triangleq \inf_{\theta_1, \theta_2 \in \mathrm{Supp}(\omega_\Theta)} \frac{D\left(X_{\theta_1}, X_{\theta_2}\right)}{R\left(X_{\theta_1}, X_{\theta_2}\right)} \\ &\leq \frac{D\left(X_{\theta_1''}, X_{\theta_2''}\right)}{R\left(X_{\theta_1''}, X_{\theta_2''}\right)} \leq \frac{D\left(X_{\theta_1'}, X_{\theta_2'}\right)}{R\left(X_{\theta_1''}, X_{\theta_2''}\right)}.\end{aligned}$$

  Therefore, we have

  $$\begin{aligned}\sup_{\theta_i \in \mathrm{Supp}(\omega_\Theta), z_i: \mathcal{M}_g(\theta_i, z_i) = \theta'} D\left(X_{\theta_1}, X_{\theta_2}\right) &= D\left(X_{\theta_1'}, X_{\theta_2'}\right) \\ \geq \gamma \cdot R\left(X_{\theta_1''}, X_{\theta_2''}\right) &> \left(\lceil \frac{1}{T} \rceil - 1\right) \cdot 2\gamma\epsilon,\end{aligned}$$

  where the last inequality utilizes the results from Eq. (8). ■

V. Data Release Mechanisms

We first present in Section V-A the *quantization mechanism*, a template for data release mechanisms used in the case studies of Section VI. The quantization mechanism can be instantiated differently for different secret functions and data distributions. We show in Section V-B techniques for instantiating the quantization mechanism, either based on theoretical insights or numerically. Finally, we give some intuition in Section V-C about how to analyze the quantization mechanism. These insights will be used in our case studies (Section VI) to show that we can sometimes match the lower bounds from Section IV up to small constant factors.

A. The Quantization Mechanism

At a high level, the quantization mechanisms follow two steps:

1) *Offline Phase:* Partition the space of parameters $\mathrm{Supp}(\Theta)$ into carefully-chosen bins.
2) *Online Phase:* For an observed data distribution parameter θ, deterministically release the quantized parameters, according to the partition from the Offline Phase.

More precisely, we first divide the set of possible distribution parameters $\mathrm{Supp}(\Theta)$ into subsets $\mathcal{S}_i$ such that $\cup_{i \in \mathcal{I}} \mathcal{S}_i \supseteq \mathrm{Supp}(\Theta)$ and $\mathcal{S}_{i_1} \cap \mathcal{S}_{i_2} = \emptyset$ for $i_1 \neq i_2$, where $\mathcal{I}$ is the (possibly uncountable) set of indices of the subsets. For $\theta \in \mathrm{Supp}(\Theta)$, $I(\theta)$ is the index of the set that θ belongs to; in other words, we have $I(\theta) = i$, where $\theta \in \mathcal{S}_i$. The mechanism first looks up which set θ belongs to (i.e., $I(\theta)$), then *deterministically* releases a parameter $\theta^*_{I(\theta)}$ that corresponds to the set. Here, θ_i^* for $i \in \mathcal{I}$ denotes another parameter. This data release mechanism has the form

$$\mathcal{M}_g(\theta, z) = \theta^*_{I(\theta)}.$$

Note that the policy is fully determined by $\mathcal{S}_i$ and θ_i^*. We will show different ways of instantiating the quantization mechanism to approach the lower bound in Section IV.

Intuitively, quantization mechanisms will have a bounded distortion as long as $d\left(\omega_{X_\theta} \| \omega_{X_{\theta^*_{I(\theta)}}}\right)$ is bounded for all $\theta \in \mathrm{Supp}(\Theta)$. At the same time, they obfuscate the secret as different data distributions within the same set are mapped to the same released parameter. This simple *deterministic* mechanism is sufficient to achieve the (order) optimal privacy-distortion trade-offs in many cases, as opposed to differential privacy, which requires randomness to provide theoretical guarantees [16] (examples in the case studies of Section VI).

B. Algorithms for Instantiating the Quantization Mechanism

To implement the quantization mechanism, we need to define the quantization bins $\mathcal{S}_i$ and the released parameter per bin θ_i^*. Depending on the data distribution, the secret function, and quantization mechanism parameters, the mechanism can have very different privacy-distortion tradeoffs. We present two methods for selecting quantization parameters: (1) an analytical approach, and (2) a numeric approach.

1) Analytical Approach (Sketch): In some cases, outlined in the case studies of Section VI and the appendices, we can find analytical expressions for $\mathcal{S}_i$ and θ_i^* while (near-)optimally trading off privacy for distortion. This is usually possible when the lower bound depends on the problem parameters in a specific way (see below). We will next illustrate the procedure through an example; precise analysis is given in Section VI.

For example, for the Gaussian distribution where $\theta = (\mu, \sigma)$, when secret=standard deviation, we can work out the lower bound from Theorem 1 (details in Appendix H in the supplementary material). Note that the lower bound is tight if our mechanism minimizes

$$\frac{D\big(X_{\mu_1,\sigma_1}, X_{\mu_2,\sigma_2}\big)}{R\big(X_{\mu_1,\sigma_1}, X_{\mu_2,\sigma_2}\big)} = \sqrt{\frac{1}{2\pi}} e^{-\frac{1}{2}\left(\frac{\mu_1-\mu_2}{\sigma_1-\sigma_2}\right)^2} - \left(\frac{\mu_1-\mu_2}{\sigma_1-\sigma_2}\right)\left(\frac{1}{2} - \Phi\left(\left(\frac{\mu_1-\mu_2}{\sigma_1-\sigma_2}\right)\right)\right), \quad (11)$$

where $D\big(X_{\theta_1}, X_{\theta_2}\big)$ and $R\big(X_{\theta_1}, X_{\theta_2}\big)$ are defined in Theorem 1, and Φ denotes the CDF of the standard Gaussian distribution. That is, for any true parameters μ_1 and σ_1, the mechanism should always choose to release μ_2 and σ_2 such that Eq. (11) is as small as possible. The exact form of Eq. (11) is not important for now; notice instead that the problem parameters (σ_i, μ_i) take the same form every time they appear in this equation. We define $t(\theta_1, \theta_2) = \frac{\mu_1-\mu_2}{\sigma_1-\sigma_2}$ to be that form.[2] Next, we find the $t(\theta_1, \theta_2)$ that minimizes Eq. (11):

$$t_0 \triangleq \arg\inf_{t(\theta_1,\theta_2)} \frac{D\big(X_{\theta_1}, X_{\theta_2}\big)}{R\big(X_{\theta_1}, X_{\theta_2}\big)}$$

For instance, in our Gaussian example, we can write t_0 as

$$t_0 = \arg\inf_{t(\theta_1,\theta_2)} \sqrt{\frac{1}{2\pi}} e^{-\frac{1}{2}(t(\theta_1,\theta_2))^2} - (t(\theta_1,\theta_2))\left(\frac{1}{2} - \Phi(t(\theta_1,\theta_2))\right),$$

which can be solved numerically. Finally, we can choose $\mathcal{S}_i$ and θ_i^* to be sets for which $t\big(\theta, \theta_i^*\big) = t_0, \ \forall \theta \in \mathcal{S}_i$. Using this rule, we derive the mechanism:

$$\mathcal{S}_{\mu,i} = \left\{\big(\mu + t_0 \cdot t, \underline{\sigma} + (i+0.5)\cdot s + t\big) | t \in \left[-\frac{s}{2}, \frac{s}{2}\right)\right\}, \quad (12)$$

$$\theta^*_{\mu,i} = \big(\mu, \underline{\sigma} + (i+0.5)\cdot s\big), \quad (13)$$

$$\mathcal{I} = \{(\mu, i) | i \in \mathbb{N}, \mu \in supp(\omega V)\}, \quad (14)$$

where s is a hyper-parameter of the mechanism that divides $\big(\overline{\sigma} - \underline{\sigma}\big)$, and $\overline{\sigma}, \underline{\sigma}$ are upper and lower bounds on σ, determined by the adversary's prior.

For our Gaussian example, the resulting sets $\mathcal{S}_{\mu,i}$ for the quantization mechanism are shown in Fig. 4; the space of possible parameters is divided into infinitely many subsets $\mathcal{S}_{\mu,i}$, each consisting of a diagonal line segment (parallel blue lines in Fig. 4). The space of possible σ values is divided into segments of length s, which correspond to the horizontal bands in Fig. 4. Given this choice of intervals, the mechanism proceeds as follows: when the true distribution parameters fall in one of these intervals, the mechanism releases the midpoint of the interval. The fact that the intervals $\mathcal{S}_{\mu,i}$ are diagonal lines arises from choosing $t(\theta_1, \theta_2) = \frac{\mu_1-\mu_2}{\sigma_1-\sigma_2}$; each interval corresponds to a set of points $(\tilde{\mu}, \tilde{\sigma})$ that satisfy $t(\theta_1, \theta_2) = t_0$, i.e., with slope $1/t_0$.

We show how to use this construction to upper bound privacy-distortion tradeoffs in Section V-C.

2) Numeric Approach: In some cases, the above procedure may not be possible. We next present a dynamic programming algorithm to numerically compute the quantization mechanism parameters. This algorithm achieves an optimal privacy-distortion tradeoff [64] among quantization algorithms with finite precision and continuous intervals $\mathcal{S}_i$. We use this algorithm (presented for univariate data distributions) in some of the case studies in Section VI.

We assume $\text{Supp}(\Theta) = \big[\underline{\theta}, \overline{\theta}\big)$, where $\underline{\theta}, \overline{\theta}$ are lower and upper bounds of θ, respectively. We consider the class of quantization mechanisms such that $\mathcal{S}_i = \Big[\underline{\theta^i}, \overline{\theta^i}\Big)$, i.e., each subset of parameters are in a continuous range. Furthermore, we explore mechanisms such that $\underline{\theta^i}, \overline{\theta^i}, \theta_i^* \in \big\{\underline{\theta}, \underline{\theta} + \kappa, \underline{\theta} + 2\kappa, \ldots, \overline{\theta}\big\}$, where κ is a hyper-parameter that encodes numeric precision (and therefore divides $(\overline{\theta} - \underline{\theta})$). For example, if we want to hide the mean of a Geometric random variable with $\underline{\theta} = 0.1$ and $\overline{\theta} = 0.9$, we could consider three-decimal-place precision, i.e., $\kappa = 0.001$ and $\underline{\theta^i}, \overline{\theta^i}, \theta_i^* \in \{0.100, 0.101, 0.102, \ldots, 0.900\}$.

Since Δ (Eq. (3)) is defined as the *worst-case* distortion whereas $\Pi_{\epsilon,\omega_\Theta}$ (Eq. (2)) is defined as a *probability*, which is related to the original data distribution, optimizing $\Pi_{\epsilon,\omega_\Theta}$ given bounded Δ (Eq. (15)) is easier to solve than the final goal of optimizing Δ given bounded $\Pi_{\epsilon,\omega_\Theta}$ (Eq. (4)).

$$\min_{\mathcal{M}_g} \ \Pi_{\epsilon,\omega_\Theta} \qquad \text{subject to } \Delta \le T. \quad (15)$$

Observing that in Eq. (4) the optimal value of $\min_{\mathcal{M}_g} \Delta$ is a monotonic decreasing function w.r.t. the threshold T, we can use a binary search algorithm (shown in Appendix C in the supplementary material) to reduce problem Eq. (4) to problem Eq. (15). It calls an algorithm that finds the optimal quantization mechanism with numerical precision over continuous intervals under a distortion budget T (i.e., solving Eq. (15)). This problem can be solved by a dynamic programming algorithm. Let $pri(t^*)$ $(t^* \in \big\{\underline{\theta}, \underline{\theta} + \kappa, \underline{\theta} + 2\kappa, \ldots, \overline{\theta}\big\})$ be the minimal privacy $\Pi_{\epsilon,\omega_\Theta}$ we can get for $\text{Supp}(\Theta) = \big\{X_\theta : \theta \in \big[\underline{\theta}, t^*\big)\big\}$ such that $\Delta \le T$. Denote $\mathcal{D}(\theta_1, \theta_2)$ as the minimal distortion a quantization mechanism can achieve under the quantization bin $[\theta_1, \theta_2)$, we have

$$\mathcal{D}(\theta_1, \theta_2) = \inf_{\theta \in \mathbb{R}^q} \sup_{\theta'' \in [\theta_1,\theta_2)} d\big(\omega_{X_{\theta''}} \| \omega_{X_\theta}\big),$$

where $d(\cdot\|\cdot)$ is defined in Eq. (3). We also denote $\mathcal{D}^*(\theta_1, \theta_2) = \arg\inf_{\theta \in [\theta_1,\theta_2)} \sup_{\theta'' \in [\theta_1,\theta_2)} d\big(\omega_{X_{\theta''}} \| \omega_{X_\theta}\big)$. If the

[2] Indeed, for many of the case studies in Section VI, $t(\theta)$ takes an analogous form; we will see the implications of this in the analysis of the upper bound in Section V-C.

prior over parameters is f_Θ, we have the Bellman equation

$$pri(t^*) = \min_{\theta\in[\underline{\theta},t^*-\kappa],\mathcal{D}(\theta,t^*)\le T} \frac{\int_{\underline{\theta}}^{\theta} f_\Theta(t)\mathrm{d}t}{\int_{\underline{\theta}}^{t^*} f_\Theta(t)\mathrm{d}t} pri(\theta) + \frac{\int_{\theta}^{t^*} f_\Theta(t)\mathrm{d}t}{\int_{\underline{\theta}}^{t^*} f_\Theta(t)\mathrm{d}t}\mathcal{P}(\theta,t^*)$$

with the initial state $pri(\underline{\theta}) = 0$, where

$$\mathcal{P}(\theta,t^*) = \mathbb{P}(\hat{g}^*(\theta') \in [g(\theta_0)-\epsilon, g(\theta_0)+\epsilon] | \theta_0 \in [\theta,t^*], \theta') = \sup_{t_1,t_2:\ \sup_{t',t''\in[t_1,t_2]}|g(t'')-g(t')|=2\epsilon} \frac{\int_{\max\{t_1,\theta\}}^{\min\{t_2,t^*\}} f_\Theta(t)\mathrm{d}t}{\int_{\theta}^{t^*} f_\Theta(t)\mathrm{d}t}.$$

θ' is the released parameter when the private parameter $\theta_0 \in [\theta, t^*]$ and $\hat{g}^*$ is the optimal attack strategy. The full algorithm is listed in Algorithm 1. The time complexity of this algorithm is $\mathcal{O}\left(\left(\overline{\theta}-\underline{\theta}/\kappa\right)^2 \cdot \mathcal{C}_D \cdot \mathcal{C}_P \cdot \mathcal{C}_I\right)$, where $\mathcal{C}_D$ is the time complexity for computing $\mathcal{D}$ and $\mathcal{D}^*$, $\mathcal{C}_P$ is the time complexity for computing $\mathcal{P}$, and $\mathcal{C}_I$ is the time complexity for computing the integrals in the Bellman equation. In our cases studies, $\mathcal{D}$ and $\mathcal{D}^*$ can be computed in $\mathcal{C}_D = \mathcal{O}(\overline{\theta}-\underline{\theta}/\kappa)$, and $\mathcal{P}$ and the integrals can be computed in closed forms within constant time, i.e., $\mathcal{C}_P = \mathcal{C}_I = \mathcal{O}(1)$.

When dynamic programming is not practical (e.g., in high-dimensional problems), we also provide a greedy algorithm in Appendix C in the supplementary material as a baseline and show the empirical comparison between these two algorithms in the case studies (Appendices F, H and I in the supplementary material).

Algorithm 1: Dynamic-Programming-Based Data Release Mechanism for Single-Parameter Distributions

Input: Parameter range: $[\underline{\theta}, \overline{\theta})$
Prior over parameter: f_Θ
Distortion budget: T
Step size: κ (which divides $\overline{\theta} - \underline{\theta}$)

```
pri(θ̲) ← 0
I(θ̲) ← ∅
for t* ← θ̲ + κ, θ̲ + 2κ, ..., θ̄ do
    pri(t*) ← ∞
    min_t ← NULL
    for θ ← t* − κ, ..., θ̲ do
        if D(θ, t*) > T then
            break
        p ← (∫_θ̲^θ f_Θ(t)dt / ∫_θ̲^t* f_Θ(t)dt) · pri(θ) + (∫_θ^t* f_Θ(t)dt / ∫_θ̲^t* f_Θ(t)dt) · P(θ, t*)
        if p < pri(t*) then
            pri(t*) ← p
            min_t ← θ
    if min_t is not NULL then
        S_t* ← [min_t, t*)
        θ'_t* ← D*(min_t, t*)
        I(t*) ← I(min_t) ∪ {t*}
if pri(θ̄) = ∞ then
    ERROR: No answer
return pri(θ̄), {S_i : i ∈ I(θ̄)}, {θ'_i : i ∈ I(θ̄)}
```

C. Technique for Analyzing the Quantization Mechanism

We next provide an overview of techniques for analyzing the quantization mechanism, both for privacy and for distortion. We use these techniques for the analysis in our case studies, where we will make the expressions and claims more precise. For concreteness, we will recall the Gaussian example from Section V-B, for which we have already derived a mechanism.

The mechanism presented in Section V-B can geometrically be interpreted as follows. Over the square of possible parameter values μ and σ (Fig. 4), the mechanism selects intervals $\mathcal{S}_{\mu,i}$ that consist of short diagonal line segments (e.g., blue line segments in Fig. 4). When the true distribution parameters fall in one of these intervals, the mechanism releases the midpoint of the interval.

We find that many of our case studies naturally give rise to the same form of $t(\theta)$. As a result, all of the case studies we analyze theoretically (with multiple parameters) have mechanisms that instantiate intervals $\mathcal{S}_{\mu,i}$ as diagonal lines, as shown in Fig. 4. The sawtooth technique, which we present next, can be used to analyze the privacy of all such mechanism instantiations. More precisely, the following pattern of quantization mechanism admits diagonal line intervals, and can be analyzed with the sawtooth technique (Section VI and Appendices F and H in the supplementary material):

$$\mathcal{S}_{\mu,i} = \left\{\left(\mu + t_0 \cdot t, \underline{\sigma} + (i+0.5)\cdot s + t\right) | t \in \left[-\frac{s}{2}, \frac{s}{2}\right)\right\},$$

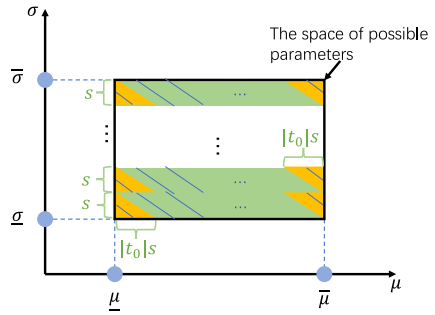


Fig. 4. We separate the space of possible parameters into two regions (yellow and green) and bound the attacker's success rate on each region separately. The blue lines represent examples of $\mathcal{S}_{\mu,i}$.

$$\theta^*_{\mu,i} = \left(\mu, \underline{\sigma} + (i+0.5)\cdot s\right),$$
$$\mathcal{I} = \{(\mu,i) | i \in \mathbb{N}, \mu \in \mathbb{R}\},$$

where s is a hyper-parameter of the mechanism that denotes quantization bin size and divides $(\overline{\sigma} - \underline{\sigma})$ and t_0 is a constant that can be determined by the mechanism design strategy described in Section V-B.

(1) Privacy analysis: For ease of illustration, we assume that the support of parameters is $\mathrm{Supp}(\Theta) = \left\{(a,b) | a \in \left[\underline{\mu}, \overline{\mu}\right), b \in \left[\underline{\sigma}, \overline{\sigma}\right)\right\}$, but the analysis can be generalized to any case.

In Fig. 4, we separate the space of possible data parameters into two regions represented by yellow and green colors. The yellow regions S_{yellow} constitute right triangles with height s and width $|t_0|s$. The green region S_{green} is the rest of the parameter space. The high-level idea of our proof is as follows. Note that for any parameter $\theta \in S_{green}$, there exists a quantization bin $\mathcal{S}_{\mu,i}$ s.t. $\theta \in \mathcal{S}_{\mu,i}$ and $\mathcal{S}_{\mu,i} \subset S_{green}$. This occurs because the mechanism intervals (blue lines in Fig. 4) all have the same slope and a length of at most s for σ. As such, each interval is either fully in the green region, or fully in the yellow region. Since we know the length of each bin, we can upper bound the attack success rate if $\theta \in S_{green}$. While the attacker can be more successful in the yellow region, the probability of $\theta \in S_{yellow}$ is small. Hence, we upper bound the overall attacker's success rate (i.e., $\Pi_{\epsilon,\omega_\Theta}$). More specifically, let the optimal attacker be $\hat{g}^*$. We have

$$\begin{aligned}\Pi_{\epsilon,\omega_\Theta} &= \mathbb{P}\big(\hat{g}^*(\theta') \in \big[g(\theta)-\epsilon, g(\theta)+\epsilon\big]\big)\\ &= \int_{\theta\in S_{green}} p(\theta)\mathbb{P}\big(\hat{g}^*(\theta') \in \big[g(\theta)-\epsilon, g(\theta)+\epsilon\big]\big)d\theta\\ &\quad + \int_{\theta\in S_{yellow}} p(\theta)\mathbb{P}\big(\hat{g}^*(\theta') \in \big[g(\theta)-\epsilon, g(\theta)+\epsilon\big]\big)d\theta\\ &< \sup_{\theta\in S_{green}} \mathbb{P}\big(\hat{g}^*(\theta') \in \big[g(\theta)-\epsilon, g(\theta)+\epsilon\big]\big)\\ &\quad + \int_{\theta\in S_{yellow}} p(\theta)d\theta\end{aligned}$$

The first term can be bounded away from 1 due to the carefully chosen t_0. The second term is bounded away from 1 because the size of S_{yellow} is relatively small. The formal justification is given in Theorem 3 and Appendices D-E2, G-B and H-D in the supplementary material.

(2) Distortion analysis. For the distortion performance, it is straightforward to show that $\Delta = \sup_{\theta\in\text{Supp}(\Theta)} d\Big(\omega_{X_\theta} \| \omega_{X_{\theta^*_{I(\theta)}}}\Big)$, where $\theta^*_{I(\theta)}$ is the released parameter when the original parameter is θ. This quantity can often be derived directly from the mechanism and parameter support.

VI. Case Studies

In this section, we instantiate the general results on concrete distributions and secrets (mean Section VI-A, quantile Section VI-B, and we defer standard deviation and discrete distribution fractions to Appendices H and I in the supplementary material). See Table I for a summary of each setting we consider, and a pointer to any theoretical results. Our results in each setting generally include a privacy lower bound, a concrete instantiation of the quantization mechanism, and privacy-distortion analysis of the data release mechanisms. In Section VI-C, we will discuss how to extend the data release mechanisms to the cases when data holders only have data samples and do not know the parameters of the underlying distributions.

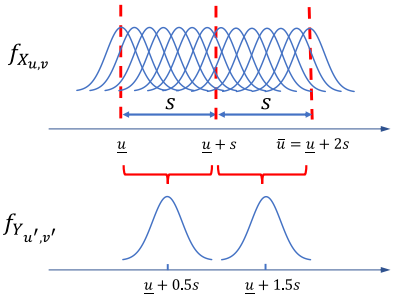


Fig. 5. Illustration of the data release mechanism for continuous distributions when secret=mean.

A. Secret = Mean

In this section, we discuss how to protect the mean of a distribution for general continuous distributions. We start with a lower bound.

Corollary 1 (Privacy lower bound, secret = mean of a continuous distribution): Consider the secret function $g(\theta) = \int_x x f_{X_\theta}(x)dx$. For any $T \in (0,1)$, when $\Pi_{\epsilon,\omega_\Theta} \le T$, we have $\Delta > \Big(\lceil \frac{1}{T} \rceil - 1\Big)\cdot\epsilon$.

The proof is in Appendix D-B in the supplementary material. We next design a data release mechanism that achieves a tradeoff close to this bound.

Data release mechanism. To begin, we restrict ourselves to continuous distributions that can be parameterized with a location parameter, where the prior distribution of the location parameter is uniform and independent of other factors:

Assumption 1: The distribution parameter vector θ can be written as (u,v), where $u \in \mathbb{R}$, $v \in \mathbb{R}^{q-1}$, and for any $u \neq u'$, $f_{X_{u,v}}(x) = f_{X_{u',v}}\big(x-u'+u\big)$. The prior over distribution parameters is $f_{U,V}(a,b) = f_U(a)\cdot f_V(b)$, where $f_U(a) = \frac{1}{\bar{u}-\underline{u}}\mathbb{I}\big(a \in [\underline{u},\bar{u})\big)$.

Examples include the Gaussian, Laplace, and uniform distributions, as well as shifted distributions (e.g., shifted exponential, shifted log-logistic). We relax this assumption to Lipschitz-continuous priors in Appendix E-A in the supplementary material, which can capture some degree of correlations between U and V. Using the strategy from Section V-B, we derive the following quantization mechanism.

Mechanism 1 (For secret = mean of a continuous distribution): The parameters of the data release mechanism are

$$\begin{aligned}&\mathcal{S}_{i,v} = \big\{(t,v)|t \in \big[\underline{u}+i\cdot s,\ \underline{u}+(i+1)\cdot s\big)\big\},\\ &\theta^*_{i,v} = \big(\underline{u}+(i+0.5)\cdot s, v\big),\\ &\mathcal{I} = \{(i,v){:}i \in \{0,1,\ldots,N-1\}, v \in supp\omega_V\},\end{aligned}$$

where s is a hyper-parameter of the mechanism that divides $\big(\bar{u}-\underline{u}\big)$ and $N = \frac{\bar{u}-\underline{u}}{s} \in \mathbb{N}$.

Fig. 5 shows an example when the original data distribution is Gaussian, i.e., $X_\theta \sim \mathcal{N}(u,v)$, and $u \in \Big[\underline{\mu},\overline{\mu}\Big)$. Intuitively, our data release mechanism "quantizes" the range of possible mean values into segments of length s. It then shifts the mean of private distribution $f_{X_{u,v}}$ to the midpoint of its corresponding segment, and releases the resulting distribution. This simple deterministic mechanism is able to achieve order-optimal privacy-distortion tradeoff in some cases, as shown below.

TABLE I
SUMMARY OF THE CASE STUDIES WE COVER, AND LINKS TO THE CORRESPONDING RESULTS

Secret \ Distribution	Continuous Distribution (order-optimal mechanism)			Ordinal Distribution (Algorithm 1)		
	Gaussian	Uniform	Exponential	Geometric	Binomial	Poisson
Mean	§VI-A			Appendix F		
Quantile	§VI-B and Appendix G			Not applicable		
Standard Deviation	Appendix H-A			Appendix H-B		
Fraction	Not applicable			Appendix I-A		

Proposition 2: Under Assumption 1, Mechanism 1 has privacy $\Pi_{\epsilon,\omega_\Theta} \le \frac{2\epsilon}{s}$ and distortion $\Delta = \frac{s}{2} < 2\Delta_{opt}(\Pi_{\epsilon,\omega_\Theta})$, where $\Delta_{opt}(\Pi_{\epsilon,\omega_\Theta})$ is the minimal distortion any data release mechanism can achieve given privacy level $\Pi_{\epsilon,\omega_\Theta}$.

The proof is in Appendix D-C in the supplementary material. The two takeaways from this proposition are that: (1) the data holder can use s to control the trade-off between distortion and privacy, and (2) the mechanism achieves an order-optimal distortion with multiplicative factor 2.

B. Secret = Quantiles

In this section, we show how to protect the α-quantile of the exponential distribution and the shifted exponential distribution. We analyze the Gaussian and uniform distributions in Appendix G in the supplementary material. We choose these distributions as the starting point of our analysis as many distributions in real-world data can be approximated by one of these distributions.

In our analysis, the parameters of (shifted) exponential distributions are denoted by:

- Exponential distribution: $\theta = \lambda$, where λ is the scale parameter: $f_{X_\lambda}(x) = \frac{1}{\lambda}e^{-x/\lambda}$.
- Shifted exponential distribution generalizes the exponential distribution with an additional shift parameter h: $\theta = (\lambda, h)$. In other words, $f_{X_{\lambda,h}}(x) = \frac{1}{\lambda}e^{-(x-h)/\lambda}$.

As before, we first present a lower bound.

Corollary 2 (Privacy lower bound, secret = α-quantile of a continuous distribution): Consider the secret function $g(\theta) = \alpha$-quantile of f_{X_θ}. For any $T \in (0, 1)$, when $\Pi_{\epsilon,\omega_\Theta} \le T$, we have $\Delta > \left(\lceil \frac{1}{T} \rceil - 1\right) \cdot 2\gamma\epsilon$, where γ is defined as follows:

- Exponential:

$$\gamma = -\frac{1}{2\ln(1-\alpha)}.$$

- Shifted exponential:

$$\gamma = \begin{cases} \frac{1}{2}\left|1 + \frac{\ln(1-\alpha)+1}{W_{-1}\left(-\frac{\ln(1-\alpha)+1}{2(1-\alpha)e}\right)}\right| & \alpha \in [0, 1-e^{-1}) \\ \frac{1}{2}\left|1 + \frac{\ln(1-\alpha)+1}{W_0\left(-\frac{\ln(1-\alpha)+1}{2(1-\alpha)e}\right)}\right| & \alpha \in [1-e^{-1}, 1) \end{cases},$$

where W_{-1} and W_0 are Lambert W functions.

The proof is given in Appendix D-D in the supplementary material. Next, we provide data release mechanisms for each of the distributions that achieve trade-offs close to these bounds.

Mechanism 2 (For secret = α-quantile of a continuous distribution): We design mechanisms for each of the distributions. In both cases, $s > 0$ is the quantization bin size chosen by the operator to divide $(\bar{\lambda} - \underline{\lambda})$, where $\bar{\lambda}$ and $\underline{\lambda}$ are upper and lower bounds of λ.

- Exponential:

$$\mathcal{S}_i = \left[\underline{\lambda} + i \cdot s, \underline{\lambda} + (i+1) \cdot s\right),$$
$$\theta_i^* = \underline{\lambda} + (i + 0.5) \cdot s,$$
$$\mathcal{I} = \mathbb{N}.$$

- Shifted exponential:

$$\mathcal{S}_{i,h} = \left\{\left(\underline{\lambda} + (i+0.5)s + t, h - t_0 \cdot t\right) | t \in \left[-\frac{s}{2}, \frac{s}{2}\right)\right\},$$
$$\theta_{i,h}^* = \left(\underline{\lambda} + (i+0.5)s, h\right),$$
$$\mathcal{I} = \{(i, h) | i \in \mathbb{N}, h \in \mathbb{R}\},$$

where

$$t_0 = \begin{cases} -1 - \ln(1-\alpha) - W_{-1}\left(-\frac{\ln(1-\alpha)+1}{2(1-\alpha)e}\right) & (\alpha \in [0, 1-e^{-1})) \\ -1 - \ln(1-\alpha) - W_0\left(-\frac{\ln(1-\alpha)+1}{2(1-\alpha)e}\right) & (\alpha \in [1-e^{-1}, 1)) \end{cases}.$$

For the privacy-distortion trade-off analysis of Mechanism 2, we assume that the parameters of the original data are drawn from a uniform distribution with lower and upper bounds. Again, we relax this assumption to Lipschitz priors in Appendix E-B in the supplementary material. Precisely,

Assumption 2: The prior over distribution parameters is:

- Exponential: λ follows the uniform distribution over $\left[\underline{\lambda}, \bar{\lambda}\right)$.
- Shifted exponential: (λ, h) follows the uniform distribution over $\left\{(a, b) | a \in \left[\underline{\lambda}, \bar{\lambda}\right), b \in \left[\underline{h}, \bar{h}\right)\right\}$.

We relax Assumption 2 and analyze the privacy-distortion trade-off of Mechanism 2 in Appendix E-B in the supplementary material.

Proposition 3: Under Assumption 2, Mechanism 2 has the following $\Pi_{\epsilon,\omega_\Theta}$ and Δ value/bound.

- *Exponential:*

$$\Pi_{\epsilon,\omega_\Theta} = \frac{2\epsilon}{-\ln(1-\alpha)s}, \qquad \Delta = \frac{1}{2}s < 2\Delta_{opt}.$$

- *Shifted exponential:*

$$\Pi_{\epsilon,\omega_\Theta} < \frac{2\epsilon}{|\ln(1-\alpha) + t_0|s} + \frac{|t_0|s}{\bar{h} - \underline{h}},$$
$$\Delta = \frac{s}{2}(t_0 - 1) + se^{-t_0}$$
$$< \left(2 + \frac{|t_0| \cdot |\ln(1-\alpha) + t_0|s^2}{\epsilon\left(\bar{h} - \underline{h}\right)}\right)\Delta_{opt}.$$

Under the high-precision regime where $\frac{s^2}{\overline{h}-\underline{h}} \to 0$ as $s, (\overline{h}-\underline{h}) \to \infty$, when $\alpha \in [0.01, 0.25] \cup [0.75, 0.99]$, Δ satisfies

$$\limsup_{\frac{s^2}{\overline{h}-\underline{h}} \to 0} \Delta < 3\Delta_{opt}.$$

Δ_{opt} is the optimal achievable distortion given the privacy achieved by Mechanism 2, and t_0 is a constant defined in Mechanism 2.

The proof is in Appendix D-E in the supplementary material. Note that the quantization bin size s cannot be too small, or the attacker can always successfully guess the secret within a tolerance ϵ (i.e., $\Pi_{\epsilon,\omega_\Theta} = 1$). Therefore, for the "high-precision" regime, we consider the asymptotic scaling as both s and $\overline{h}-\underline{h}$ grow. When $s > 1$, the scaling condition $\frac{s^2}{\overline{h}-\underline{h}} \to 0$ implies a more interpretable condition of $\frac{s}{\overline{h}-\underline{h}} \to 0$, which says that the bin size is small relative to the parameter space. For example, this condition is required when the secret tolerance $\epsilon > 1/2$ (i.e., we need a bin size $s > 1$ to achieve non-trivial privacy guarantees).

Proposition 3 shows that the quantization mechanism is order-optimal with multiplicative factor 2 for the exponential distribution. For shifted exponential distribution, order-optimality holds asymptotically in the high-precision regime.

C. Extending Data Release Mechanisms for Dataset Inputs and Outputs

The data release mechanisms discussed in previous sections assume that data holders know the *distribution parameter* of the original data. In practice, data holders often only have a dataset of samples from the data distribution and do not know the parameters of the underlying distributions. Quantization data release mechanisms can be easily adapted to handle dataset inputs and outputs.

The high-level idea is that the data holders can estimate the distribution parameters θ from the data samples and find the corresponding quantization bins $\mathcal{S}_i$ according to the estimated parameters, and then modify the original samples as if they were sampled according to the released parameter θ_i^*. This may be infeasible for high-dimensional parameter vectors θ; we did not explore this question in the current work. For brevity, we only present the concrete procedure for secret=mean in continuous distributions as an example. For a dataset of $\mathcal{X} = \{x_1, \ldots, x_n\}$, the procedure is the following.

1) Estimate the mean from the data samples: $\hat{\mu} = \frac{1}{n}\sum_{i\in[n]} x_i$.
2) According to Eq. (16), compute the index of the corresponding set $i = \lfloor \frac{\hat{\mu}-\underline{\mu}}{s} \rfloor$.
3) According to Eq. (13), change the mean of the data samples to $\mu_{target} = \underline{\mu} + (i + 0.5) \cdot s$. This can be done by a sample-wise operation $x_i' = x_i - \hat{\mu} + \mu_{target}$.
4) The released dataset is $\mathcal{M}_g(\mathcal{X}, z) = \{x_1', \ldots, x_n'\}$.

Note that this mechanism applies to samples. Therefore, it can be applied either to the original data, or as an add-on to existing data sharing tools [15], [65], [66], [67], [68]. For example, it can be used to modify synthetically-generated samples after they are generated, or to modify the training dataset for a generative model, or to directly modify the original data for releasing.

VII. Experiments

In the previous sections, we theoretically demonstrated the privacy-distortion tradeoffs of our data release mechanisms in some special case studies. In this section, we focus on *orthogonal* questions through real-world experiments: (1) how well our data release mechanisms perform in practice when the assumptions do not hold, and (2) how summary statistic privacy quantitatively compares with existing privacy frameworks (which we explained qualitatively in Section II).[3]

Datasets. We use two real-world datasets to simulate the motivating scenarios.

1) Wikipedia Web Traffic Dataset (WWT) [69] contains the daily page views of 145,063 Wikipedia web pages in 2015-2016. To preprocess it for our experiments, we remove the web pages with empty page view record on any day (117,277 left), and compute the mean page views across all dates for each web page. Our goal is to release the page views (i.e., a 117,277-dimensional vector) while protecting the **mean of the distribution** (which reveals the business scales of the company).
2) Measuring Broadband America Dataset (MBA) [70] contains network statistics (including network traffic counters) collected by United States Federal Communications Commission from homes across United States. We select the average network traffic (GB/measurement) from AT&T clients as our data. Our goal is to release a copy of this data while hiding the **0.95-quantile** (which reveals the network capability).

Baselines. We compare our mechanisms discussed in Section VI with three popular mechanisms proposed in prior work (Section II): differentially-private density estimation [23] (shortened to DP), attribute-private Gaussian mechanism [18] (shortened to AP), and Wasserstein mechanism for distribution privacy [28] (shortened to DistP). As these mechanisms provide different privacy guarantees than summary statistic privacy, it is difficult to do a fair comparison between these baselines and our quantization mechanism. We include them to quantitatively show the differences (and similarities) between various privacy frameworks.

For a dataset of samples $\mathcal{X} = \{x_1, \ldots, x_n\}$, DP works by: (1) Dividing the space into m bins: $B_1, \ldots, B_m$. (2) Computing the histogram $C_i = \sum_{j=1}^n \mathbb{I}(x_j \in B_i)$. (3) Adding noise to the histograms $D_i = \max\{0, C_i + \text{Laplace}(0, \beta^2)\}$, where $\text{Laplace}(0, \beta^2)$ means a random noise from Laplace distribution with mean 0 and variance β^2. (4) Normalizing the histogram $p_i = \frac{D_i}{\sum_{j=1}^m D_j}$. We can then draw y_i according to the histogram and release $\mathcal{Y} = \{y_1, \ldots, y_n\}$ with differential privacy guarantees. AP works by releasing $\mathcal{Y} = \{x_i + \mathcal{N}(0, \beta^2)\}_{i=1}^n$. DistP works by releasing $\mathcal{Y} =$

[3]Code available at https://github.com/fjxmlzn/summary_statistic_privacy.

$\{x_i + \text{Laplace}(0, \beta^2)\}_{i=1}^n$. Note that for each of these mechanisms, normally their noise parameters would be set carefully to match the desired privacy guarantees (e.g., differential privacy). In our case, since our privacy metric is different, it is unclear how to set the noise parameters for a fair privacy comparison. For this reason, we evaluate different settings of the noise parameters, and measure the empirical tradeoffs.

Metrics. Our privacy and distortion metrics depend on the prior distribution of the original data $\theta \sim \omega_\Theta$ (though the mechanism does not). In practice (and also in these experiments), the data holder only has one dataset. Therefore, we cannot empirically evaluate the proposed privacy and distortion metrics, and resort to surrogate metrics to bound our true privacy and distortion.

Surrogate privacy metric. For an original dataset $\mathcal{X} = \{x_1, \ldots, x_n\}$ and the released dataset $\mathcal{Y} = \{y_1, \ldots, y_n\}$, we define the surrogate privacy metric $\tilde{\Pi}_\epsilon$ as the error of an attacker who guesses the secret of the released dataset as the true secret: $\tilde{\Pi}_\epsilon \triangleq -|g(\mathcal{X}) - g(\mathcal{Y})|$, where $g(\mathcal{D})$ = mean of $\mathcal{D}$ and 0.95-quantile of $\mathcal{D}$ in WWT and MBA datasets respectively. Note that in the definition of $\tilde{\Pi}_\epsilon$, a minus sign is added so that a smaller value indicates stronger privacy, as in privacy metric Eq. (2). This simple attacker strategy is in fact a good proxy for evaluating the privacy $\Pi_{\epsilon,\omega_\Theta}$ due to the following facts. (1) For our data release mechanisms for these secrets Mechanism 1,Mechanism 2, when the prior distribution is uniform, this strategy is actually optimal, so there is a direct mapping between $\tilde{\Pi}_\epsilon$ and $\Pi_{\epsilon,\omega_\Theta}$.

(2) For AP applied on protecting mean of the data (i.e., Wikipedia Web Traffic Dataset experiments), this strategy gives an unbiased estimator of the secret. (3) For DP and AP on other cases, this mechanism may not be an unbiased estimator of the secret, but it gives an *upper bound* on the attacker's error.

Surrogate distortion metric. We define our surrogate distortion metric as the Wasserstein-1 distance between the two datasets: $\tilde{\Delta} \triangleq d(p_{\mathcal{X}} \| p_{\mathcal{Y}})$ where p_D denotes the empirical distribution of a dataset D. This metric evaluates how much the mechanism distorts the dataset.

In fact, we can deduce a theoretical lower bound for the surrogate privacy and distortion metrics for secret = mean (shown later in Fig. 6) using similar techniques as the proofs in the main paper (see Appendix D-F in the supplementary material).

A. Results

We enumerate the hyper-parameters of each method (bin size and β for DP, β for AP and DistP, and s for ours). For each method and each hyper-parameter, we compute their surrogate privacy and distortion metrics. The results are shown in Fig. 6 (bottom left is best); each data point represents one realization of mechanism $\mathcal{M}_g$ under a distinct hyperparameter setting. Two takeaways are below.

(1) The proposed quantization data release mechanisms has a good surrogate privacy-distortion trade-off, even when the assumptions do not hold. The data distributions analyzed in Section VI and in the Appendices may not always match real data exactly. Our data release mechanism for mean (i.e., Mechanism 1 used in WWT) supports general continuous distributions. Indeed, even for our surrogate metrics, our Mechanism 1 is also optimal (see Appendix D-F in the supplementary material, Fig. 6(a)). However, the quantization data release mechanisms for quantiles (i.e., Mechanism 2 used in Fig. 6(b)) are order-optimal only when the distributions are within certain classes (Section VI-B). Since network traffic in MBA is one-sided and heavy-tailed, we use the data release mechanism for exponential distributions (Mechanism 2), which is not heavy-tailed. Despite the distribution mismatch, the quantization data release mechanism still achieves a good (surrogate) privacy-distortion compared to DP, AP, and DistP (Fig. 6(b)).

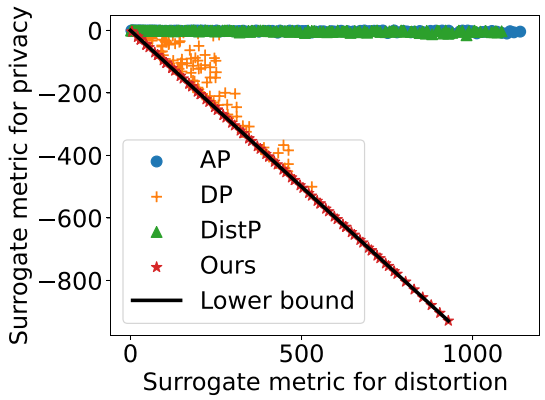
(2) The quantization data release mechanisms achieve better privacy-distortion trade-off than DP, AP, and DistP. AP and DistP directly add Gaussian/Laplace noise to each sample. This process does not change the mean of the distribution on expectation. Therefore, Figure 6 shows that AP and DistP have a bad privacy-distortion tradeoff. DP quantizes (bins) the samples before adding noise. Quantization has a better property in terms of protecting the mean of the distribution, and therefore we see that DP has a better privacy-distortion tradeoff than AP and DistP, but still worse than the quantization mechanism. Note that in Fig. 6(b), a few of the DP instances have better privacy-distortion trade-offs than ours. This is not an indication that DP is fundamentally better. Due to the randomness in DP (from the added Laplace noise), some realizations of the noise in this experiment give a better trade-off. Another instance of the DP algorithm could give a worse trade-off, so DP's achievable trade-off points are widespread.

In summary, these empirical results confirm the intuition in Section II that DP, AP, and DistP may not achieve good privacy-utility tradeoffs for our problem. This is expected—they are designed for a different objective. Additional results on downstream tasks are in Appendix J in the supplementary material.

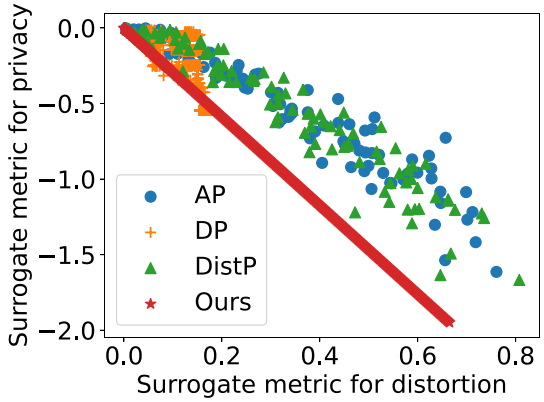
VIII. Discussion and Future Work

This work introduces a framework for *summary statistic privacy* concerns in data sharing applications. This framework can be used to analyze the leakage of statistical information and the privacy-distortion trade-offs of data release mechanisms (Sections III and IV). The quantization data release mechanisms can be used to protect statistical information (Sections V and VI). However, many interesting open questions for future work remain.

Number of secrets. In this work, we studied the case where the data holder only wishes to hide a single secret. In practice, data holders often want to hide multiple properties of their underlying data. The challenges of studying this setting are twofold. The first is *metric design*. Although we can adopt the same high level ideas of designing privacy and distortion metrics in this paper, the data holders' tolerances (ϵ in the current work) can differ for different secrets (e.g., from not allowing any secret to be disclosed, to tolerating at most a small subset of secrets being revealed). It is not clear which



(a) Wikipedia Web Traffic Dataset. (secret=mean)



(b) Measuring Broadband America Dataset. (secret=quantile)

Fig. 6. Privacy (lower is better) and distortion (lower is better) of AP, DP, DistP, and ours. Each point represents one instance of data release mechanism with one hyper-parameter. "Lower bound" is the theoretical lower bound of the achievable region. Our data release mechanisms achieve better privacy-distortion tradeoff than AP, DP, and DistP.

of these operating points is most practically relevant and analytically tractable. Another challenge is *tradeoff analysis*. With multivariate secrets and different data release goals, the analysis of theoretical privacy-distortion tradeoff lower bounds may require different analysis techniques.

The dimension and the type of data distributions. Although the proof for the lower bound in Section IV applies to general prior distributions, we analyze the quantization mechanism under a limited set of one-dimensional distributions (Table I), assuming different parameters of the distribution are drawn independently (Section VI) or follow a Lipschitz-continuous prior, (Appendices E, G-C and H-E in the supplementary material), which can capture some degree of correlation. An interesting direction for future work is to design mechanisms that have good tradeoffs under prior distributions with arbitrarily correlated parameters.

Relation to Differential Privacy Figure 6 suggests that despite being designed for a different threat model, the DP mechanism does fairly well. As mentioned, this is because the mechanism first bins data points, which is similar to quantization. However, this raises an important question: under what conditions on the true data, the secret quantity, and the mechanism do differentially-private mechanisms achieve a good privacy-utility tradeoff for our problem?

Robustness to practical factors. Our privacy metric $\Pi_{\epsilon,\omega_\Theta}$ requires knowledge of the prior distribution of the parameters ω_Θ. In practice, however, the data holders' estimated prior $\hat{\omega}_\Theta$ may not match the ground truth. A natural question is how this mismatch affects the data holder's privacy guarantees. To this end, we first define a notion of robustness for our privacy metric, and show that Mechanism 1 is *not* robust.

Definition 1 (Robustness to Prior Misspecification): Let ω_Θ be the true prior distribution of the parameters, $\hat{\omega}_\Theta$ be the data holders' estimated prior, and $d_{\omega_\Theta,\hat{\omega}_\Theta}$ be the Wasserstein-1 distance between ω_Θ and $\hat{\omega}_\Theta$. For a mechanism $\mathcal{M}_g$, the privacy of it is $\Pi_{\epsilon,\omega_\Theta}$, while the data holder miscalculate the privacy as $\Pi_{\epsilon,\hat{\omega}_\Theta}$. $\mathcal{M}_g$ is r-robust if for any $\epsilon > 0$, ω_Θ, and $\hat{\omega}_\Theta$,

$$\frac{\Pi_{\epsilon,\omega_\Theta}}{\Pi_{\epsilon,\hat{\omega}_\Theta}} \leq 1 + r d_{\omega_\Theta,\hat{\omega}_\Theta}.$$

Proposition 4: There is no constant $r < \infty$ for which Mechanism 1 is r-robust.

The proof is in Appendix D-G in the supplementary material. This result shows a weakness of the quantization mechanism, but more importantly, it highlights the fragility of the privacy metric studied in this work. Min-entropy (and variants thereof), while a natural and interpretable privacy metric in some respects, admits solutions like the quantization mechanism that are fragile to prior misspecification.

A related type of fragility is that the privacy metric we study does not provide composition guarantees; in other words, if one applies a summary statistic-private mechanism υ times, we cannot easily bound the privacy parameter of the υ-fold composed mechanism. In contrast, composition is an important and desirable property exhibited by differential privacy [16]. The lack of composition can be problematic in situations where a data holder wants to release a dataset (or correlated datasets) multiple times.

Future direction: New privacy metrics. Several of the previously-discussed challenges, especially related to robustness, stem from our choice of privacy metric. It remains an open problem to design privacy metrics that can protect trade secrets while (a) not requiring knowledge of the data prior, (b) providing composition guarantees, and (c) not adding excessive noise. Motivated by maximal leakage [20], one could consider a *normalized* variant of the studied privacy metric:

$$\Pi'_{\epsilon,\omega_\Theta} \triangleq \sup_{\omega_\Theta} \log \frac{\Pi_{\epsilon,\omega_\Theta}}{\sup_{\hat{g}} \mathbb{P}\big(\hat{g}(\omega_\Theta) \in [g(\theta)-\epsilon, g(\theta)+\epsilon]\big)},$$

where $\hat{g}(\omega_\Theta)$ is an attacker that knows the prior distribution but does not see the released data, and the denominator is the probability that the strongest attacker guesses the secret within tolerance ϵ. Similar to maximal leakage, we consider the worst-case leakage among all possible priors. This *normalized*

$\Pi'_{\epsilon,\omega_\Theta}$ considers how much additional "information" that the released data provides to the attacker in the worst-case (see also inferential privacy [71]).

REFERENCES

[1] H. L. Lee and S. Whang, "Information sharing in a supply chain," *Int. J. Manuf. Technol. Manage.*, vol. 1, no. 1, pp. 79–93, 2000.

[2] N. Choucri, S. Madnick, and P. Koepke, *Institutions for Cyber Security: International Responses and Data Sharing Initiatives*. Cambridge, MA, USA: Massachusetts Inst. Technol., 2016.

[3] J. B. Jacobs and D. Blitsa, "Sharing criminal records: The United States, the european union and interpol compared," *Loyola Los Angeles Int. Comput. Law Rev.*, vol. 30, p. 125, Jan. 2008.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE CVPR*, 2009, pp. 248–255.

[5] C. Reiss, J. Wilkes, and J. L. Hellerstein, "Google cluster-usage traces: Format+ schema," Google Inc., Mountain View, CA, USA, White Paper, pp. 1–14, 2011.

[6] S. Luo et al., "Characterizing microservice dependency and performance: Alibaba trace analysis," in *Proc. ACM SoCC*, 2021, pp. 412–426.

[7] A. Suri and D. Evans, "Formalizing and estimating distribution inference risks," 2021, *arXiv:2109.06024*.

[8] A. Suri, Y. Lu, Y. Chen, and D. Evans, "Dissecting distribution inference," in *Proc. 1st IEEE Conf. Secure Trustworthy Mach. Learn.*, 2023, pp. 1–16.

[9] G. Ateniese, L. V. Mancini, A. Spognardi, A. Villani, D. Vitali, and G. Felici, "Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers," *Int. J. Security Netw.*, vol. 10, no. 3, pp. 137–150, 2015.

[10] K. Ganju, Q. Wang, W. Yang, C. A. Gunter, and N. Borisov, "Property inference attacks on fully connected neural networks using permutation invariant representations," in *Proc. ACM SIGSAC Conf. Comput. Commun. Security*, 2018, pp. 619–633.

[11] W. Zhang, S. Tople, and O. Ohrimenko, "Leakage of Dataset properties in multi-party machine learning," in *Proc. USENIX Security Symp.*, 2021, pp. 2687–2704.

[12] S. Mahloujifar, E. Ghosh, and M. Chase, "Property inference from poisoning," in *Proc. Security Privacy*, 2022, pp. 1–18.

[13] H. Chaudhari, J. Abascal, A. Oprea, M. Jagielski, F. Tramèr, and J. Ullman, "SNAP: Efficient extraction of private properties with poisoning," in *Proc. IEEE SP*, 2022, pp. 1935–1952.

[14] B. Imana, A. Korolova, and J. Heidemann, "Institutional privacy risks in sharing DNS data," in *Proc. ANRW*, 2021, pp. 69–75.

[15] Z. Lin, A. Jain, C. Wang, G. Fanti, and V. Sekar, "Using GANs for sharing networked time series data: Challenges, initial promise, and open questions," in *Proc. ACM IMC*, 2020, pp. 464–483.

[16] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. TCC*, New York, NY, USA, Mar. 2006, pp. 265–284.

[17] C. Reiss, J. Wilkes, and J. L. Hellerstein, "Obfuscatory obscanturism: Making workload traces of commercially-sensitive systems safe to release," in *Proc. IEEE NOMS*, 2012, pp. 1279–1286.

[18] W. Zhang, O. Ohrimenko, and R. Cummings, "Attribute privacy: Framework and mechanisms," in *Proc. FACCT*, 2022, pp. 1–20.

[19] A. Makhdoumi, S. Salamatian, N. Fawaz, and M. Médard, "From the information bottleneck to the privacy funnel," in *Proc. IEEE ITW*, 2014, pp. 501–505.

[20] I. Issa, A. B. Wagner, and S. Kamath, "An operational approach to information leakage," *IEEE Trans. Inf. Theory*, vol. 66, no. 3, pp. 1625–1657, Mar. 2020.

[21] H. Wang, L. Vo, F. P. Calmon, M. Médard, K. R. Duffy, and M. Varia, "Privacy with estimation guarantees," *IEEE Trans. Inf. Theory*, vol. 65, no. 12, pp. 8025–8042, Dec. 2019.

[22] S. Asoodeh, M. Diaz, F. Alajaji, and T. Linder, "Privacy-aware guessing efficiency," in *Proc. ISIT*, 2017, pp. 1–5.

[23] L. Wasserman and S. Zhou, "A statistical framework for differential privacy," *J. Amer. Stat. Assoc.*, vol. 105, no. 489, pp. 375–389, 2010.

[24] B. Bebensee, "Local differential privacy: A tutorial," 2019, *arXiv:1907.11908*.

[25] K. Chatzikokolakis, M. E. Andrés, N. E. Bordenabe, and C. Palamidessi, "Broadening the scope of differential privacy using metrics," in *Proc. PETS*, Bloomington, IN, USA, Jul. 2013, pp. 82–102.

[26] D. Kifer and A. Machanavajjhala, "Pufferfish: A framework for mathematical privacy definitions," *ACM Trans. Database Syst.*, vol. 39, no. 1, pp. 1–36, 2014.

[27] Y. Kawamoto and T. Murakami, "Local obfuscation mechanisms for hiding probability distributions," in *Proc. Comput. Security*, 2019, pp. 128–148.

[28] M. Chen and O. Ohrimenko, "Protecting global properties of datasets with distribution privacy mechanisms," in *Proc. AISTATS*, 2023, pp. 1–20.

[29] H. Yamamoto, "A source coding problem for sources with additional outputs to keep secret from the receiver or wiretappers (corresp.)," *IEEE Trans. Inf. Theory*, vol. 29, no. 6, pp. 918–923, Nov. 1983.

[30] G. Smith, "On the foundations of quantitative information flow," in *Proc. FoSSaCS*, 2009, pp. 288–302.

[31] M. S. Alvim, K. Chatzikokolakis, A. McIver, C. Morgan, C. Palamidessi, and G. Smith, "Additive and multiplicative notions of leakage, and their capacities," in *Proc. IEEE CSF*, 2014, pp. 308–322.

[32] S. Asoodeh, M. Diaz, F. Alajaji, and T. Linder, "Information extraction under privacy constraints," *Information*, vol. 7, no. 1, p. 15, 2016.

[33] J. Liao, O. Kosut, L. Sankar, and F. du Pin Calmon, "Tunable measures for information leakage and applications to privacy-utility tradeoffs," *IEEE Trans. Inf. Theory*, vol. 65, no. 12, pp. 8043–8066, Dec. 2019.

[34] F. P. Calmon, A. Makhdoumi, and M. Médard, "Fundamental limits of perfect privacy," in *Porc. ISIT*, 2015, pp. 1796–1800.

[35] S. Saeidian, G. Cervia, T. J. Oechtering, and M. Skoglund, "Pointwise maximal leakage," in *Proc. IEEE ISIT*, 2022, pp. 626–631.

[36] G. R. Kurri, L. Sankar, and O. Kosut, "An operational approach to information leakage via generalized gain functions," 2022, *arXiv:2209.13862*.

[37] A. Gilani, G. R. Kurri, O. Kosut, and L. Sankar, "Unifying privacy measures via maximal (α, β)-leakage (MαbeL)," *IEEE Trans. Inf. Theory*, vol. 70, no. 6, pp. 4368–4395, Jun. 2024.

[38] M. Jurado, R. G. Gonze, M. S. Alvim, and C. Palamidessi, "Analyzing the shuffle model through the lens of quantitative information flow," 2023, *arXiv:2305.13075*.

[39] R. Jin, X. He, and H. Dai, "On the security-privacy tradeoff in collaborative security: A quantitative information flow game perspective," *IEEE Trans. Inf. Forensics Security*, vol. 14, pp. 3273–3286, 2019.

[40] M. S. Alvim, N. Fernandes, A. McIver, C. Morgan, and G. H. Nunes, "A novel analysis of utility in privacy pipelines, using Kronecker products and quantitative information flow," in *Proc. ACM CCS*, 2023, pp. 1718–1731.

[41] R. Bhaskar, A. Bhowmick, V. Goyal, S. Laxman, and A. Thakurta, "Noiseless database privacy," in *Proc. ASIACRYPT*, Seoul, South Korea, Dec. 2011, pp. 215–232.

[42] F. Farokhi, "Development and analysis of deterministic privacy-preserving policies using non-stochastic information theory," *IEEE Trans. Inf. Forensics Security*, vol. 14, pp. 2567–2576, 2019.

[43] F. Farokhi, "Noiseless privacy: Definition, guarantees, and applications," *IEEE Trans. Big Data*, vol. 9, no. 1, pp. 51–62, Feb. 2023.

[44] F. Farokhi and G. Nair, "Non-stochastic private function evaluation," in *Proc. IEEE ITW*, 2021, pp. 1–5.

[45] B. Rassouli and D. Gündüz, "On perfect privacy," *IEEE J. Sel. Areas Inf. Theory*, vol. 2, no. 1, pp. 177–191, Mar. 2021.

[46] A. Zamani, T. J. Oechtering, and M. Skoglund, "Bounds for privacy-utility trade-off with non-zero leakage," in *Proc. IEEE ISIT*, 2022, pp. 620–625.

[47] S. Biswas and C. Palamidessi, "PRIVIC: A privacy-preserving method for incremental collection of location data," in *Proc. Privacy Enhanc. Technol.*, 2024, pp. 1–15.

[48] S. Oya, C. Troncoso, and F. Pérez-González, "Back to the drawing board: Revisiting the design of optimal location privacy-preserving mechanisms," in *Proc. ACM CCS*, 2017, pp. 1959–1972.

[49] D. Clark, S. Hunt, and P. Malacaria, "Quantitative analysis of the leakage of confidential data," *Electron. Notes Theor. Comput. Sci.*, vol. 59, no. 3, pp. 238–251, 2002.

[50] D. Clark, S. Hunt, and P. Malacaria, "A static analysis for quantifying information flow in a simple imperative language," *J. Comput. Security*, vol. 15, no. 3, pp. 321–371, 2007.

[51] P. Malacaria, "Assessing security threats of looping constructs," in *Proc. ACM POPL*, 2007, pp. 225–235.

[52] B. Rassouli and D. Gündüz, "Optimal utility-privacy trade-off with total variation distance as a privacy measure," *IEEE Trans. Inf. Forensics Security*, vol. 15, pp. 594–603, 2019.

[53] S. A. Mario, K. Chatzikokolakis, C. Palamidessi, and G. Smith, "Measuring information leakage using generalized gain functions," in *Proc. CSF*, 2012, pp. 265–279.

[54] S. Asoodeh, M. Diaz, F. Alajaji, and T. Linder, "Estimation efficiency under privacy constraints," *IEEE Trans. Inf. Theory*, vol. 65, no. 3, pp. 1512–1534, Mar. 2019.
[55] N. E. Bordenabe and G. Smith, "Correlated secrets in quantitative information flow," in *Proc. CSF*, 2016, pp. 93–104.
[56] C. E. Shannon, "Coding theorems for a discrete source with a fidelity criterion," *IRE Int. Conv. Rec.*, vol. 4, nos. 142–163, p. 1, 1959.
[57] S. Arimoto, "An algorithm for computing the capacity of arbitrary discrete memoryless channels," *IEEE Trans. Inf. Theory*, vol. IT-18, no. 1, pp. 14–20, Jan. 1972.
[58] R. Blahut, "Computation of channel capacity and rate-distortion functions," *IEEE Trans. Inf. Theory*, vol. 18, no. 4, pp. 460–473, Jan. 1972.
[59] D. E. R. Denning, *Cryptography and Data Security*, vol. 112. Reading, MA, USA: Addison-Wesley, 1982.
[60] J. W. Gray, III, "Toward a mathematical foundation for information flow security," *J. Comput. Security*, vol. 1, nos. 3–4, pp. 255–294, 1992.
[61] G. N. Nair, "A nonstochastic information theory for communication and state estimation," *IEEE Trans. Autom. Control*, vol. 58, no. 6, pp. 1497–1510, Jun. 2013.
[62] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein generative adversarial networks," in *Proc. ICML*, 2017, pp. 214–223.
[63] Z. Lin, A. Khetan, G. Fanti, and S. Oh, "PacGAN: The power of two samples in generative adversarial networks," in *Proc. NeurIPS*, 2018, pp. 1–10.
[64] R. Bellman, "Dynamic programming," *Science*, vol. 153, no. 3731, pp. 34–37, 1966.
[65] C. Esteban, S. L. Hyland, and G. Rätsch, "Real-valued (medical) time series generation with recurrent conditional GANs," 2017, *arXiv:1706.02633*.
[66] Y. Yin, Z. Lin, M. Jin, G. Fanti, and V. Sekar, "Practical GAN-based synthetic IP header trace generation using NetShare," in *Proc. ACM SIGCOMM*, 2022, pp. 458–472.
[67] J. Jordon, J. Yoon, and M. Van Der Schaar, "PATE-GAN: Generating synthetic data with differential privacy guarantees," in *Proc. ICLR*, 2018, pp. 1–21.
[68] J. Yoon, D. Jarrett, and M. Van der Schaar, "Time-series generative adversarial networks," in *Proc. NeurIPS*, vol. 32, 2019, pp. 1–11.
[69] "Web traffic time series forecasting," Google. 2018. [Online]. Available: https://www.kaggle.com/c/web-traffic-time-series-forecasting
[70] (Federal Commun. Comm., Washington, DC, USA). *Raw Data—Measuring Broadband America—Seventh Report*. (2018). [Online]. Available: https://www.fcc.gov/reports-research/reports/measuring-broadband-america/raw-data-measuring-broadband-america-seventh
[71] A. Ghosh and R. Kleinberg, "Inferential privacy guarantees for differentially private mechanisms," in *Proc. ITCS*, 2017, pp. 1–31.

Zinan Lin received the B.E. degree from Tsinghua University in 2017, and the Ph.D. degree from Carnegie Mellon University in 2023. He is a Senior Researcher with Microsoft Research. His research interests are generative modeling and privacy. His work has been recognized with several awards, including the Outstand Paper/Oral/Spotlight Awards at NeurIPS and the Best Paper Finalist at IMC.

Shuaiqi Wang (Graduate Student Member, IEEE) received the B.E. degree from Shanghai Jiao Tong University in 2020. He is currently pursuing the Ph.D. degree with Carnegie Mellon University. His work has been recognized by the Carnegie Institute of Technology Dean's Fellow. His research interests are data privacy and security in machine learning.

Vyas Sekar received the B.Tech. degree from the Indian Institute of Technology Madras (IIT Madras), and the Ph.D. degree from Carnegie Mellon University, Pittsburgh, where he is a Tan Family Professor with the ECE Department. He also serves as the Chief Scientist with Conviva, and as a Cofounder with Rockfish Data, a startup commercializing his academic research on synthetic data. His work has been recognized with numerous awards, including the SIGCOMM Rising Star Award, the SIGCOMM Test of Time Award, the NSA Science of Security Prize, the NSF CAREER Award, the Internet Research Task Force Applied Networking Research Award, the Intel Outstanding Researcher Award, and the IIT Madras Young Alumni Achiever Award. He was awarded the President of India Gold Medal from IIT Madras.

Giulia Fanti (Member, IEEE) received the B.S. degree in ECE from the Olin College of Engineering, and the Ph.D. degree in EECS from the University of California at Berkeley. She is an Assistant Professor of Electrical and Computer Engineering with Carnegie Mellon University. Her research interests span the security, privacy, and efficiency of distributed systems. She is a two-time Fellow of the World Economic Forum's Global Future Council on Cybersecurity and a member of NIST's Information Security and Privacy Advisory Board. Her work has been recognized with several awards, including the best paper awards, a Sloan Fellowship, an Intel Rising Star Faculty Award, and an ACM SIGMETRICS Rising Star Award.