SoK: Descriptive Statistics Under Local Differential Privacy

René Raab[1], Pascal Berrang[2], Paul Gerhart[3], and Dominique Schröder[2,3]

[2] *University of Birmingham*
[1] *Friedrich-Alexander-Universität Erlangen-Nürnberg*
[3] *TU Wien*

Abstract

Local Differential Privacy (LDP) provides a formal guarantee of privacy that enables the collection and analysis of sensitive data without revealing any individual's data. While LDP methods have been extensively studied, there is a lack of a systematic and empirical comparison of LDP methods for descriptive statistics. In this paper, we first provide a systematization of LDP methods for descriptive statistics, comparing their properties and requirements. We demonstrate that several mean estimation methods based on sampling from a Bernoulli distribution are equivalent in the one-dimensional case and introduce methods for variance estimation. We then empirically compare methods for mean, variance, and frequency estimation. Finally, we provide recommendations for the use of LDP methods for descriptive statistics and discuss their limitations and open questions.

1 Introduction

The advent of mobile applications has led to the emergence of numerous modern applications that necessitate the collection and analysis of sensitive data generated in a decentralized manner by different users or devices. These include, but are not limited to, applications in the field of medicine [Raa+23; Kai+20; Amm+23; Zie+20], telemetry [DKY17], or usage statistics [EPK14; Dif17]. Historically, this data has been collected by a central entity for analysis, which has required individuals to trust the central entity. In light of recent advances towards sharing sensitive data with more entities, especially in the field of healthcare, the consideration of individuals' privacy has become more critical.

While these advances in data sharing often require the use of anonymization or pseudonymization techniques, these methods have been demonstrated to be inadequate for the protection of privacy, as evidenced by the findings of Sweeney [Swe97] and Berrang, Gerhart, and Schröder [BGS24]. Differential Privacy (DP), as proposed by Dwork, represents a more robust privacy definition that provides strong privacy guarantees, yet still requires trust in the central entity [Dwo+06]. The local variant of differential privacy, Local Differential Privacy (LDP) [Kas+11], eliminates the need for trust in the system. In LDP, each participant perturbs their data locally before sending it to the central aggregator. The LDP mechanism ensures a formal privacy guarantee for each user while

enabling the central entity to estimate aggregate statistics. LDP preserves privacy by introducing noise into the data shared with the aggregator, making it difficult to perform accurate analysis. This privacy-utility trade-off is a central challenge in the design of LDP mechanisms.

A review of the literature reveals that there are only a few documented applications of LDP in practice, with the majority of these being implemented by large corporations [DKY17; EPK14; Dif17]. In contrast, the majority of the literature focuses on theoretical aspects of LDP. This underscores a significant disparity between LDP research and its practical applications, likely attributable to the absence of a comprehensive overview and empirical comparison to assess the efficacy of LDP methods in real-world settings.

A number of surveys on LDP methods have been conducted, e.g., [Xio+20; Wan+20; Yan+24]. However, none of these surveys provide a comprehensive empirical evaluation of these methods for data analysis, which are crucial for practical implementation. These evaluations reveal real-world performance, context-specific effectiveness, and implementation challenges that theoretical analyses often overlook. Empirical studies serve to bridge the gap between theory and practice, thereby aiding practitioners in making informed decisions and driving further research and development. For instance, Wang et al. [Wan+20] review various LDP techniques, focusing on definitions, frequency estimation, mean estimation, and machine learning. However, their work does not offer empirical comparisons. Similarly, Xiong et al. [Xio+20] summarize LDP applications in frequency estimation, mean estimation, distribution estimation, and machine learning. However, their work also lacks empirical evaluations. Yang et al. [Yan+24] present an overview of LDP methods and applications, but their work provides only a broad summary. They do not cover all methods comprehensively and do not emphasize descriptive statistics. Consequently, a detailed empirical comparison of LDP methods remains a critical missing piece in the literature, necessary for bridging the gap between theoretical research and practical application.

This paper presents a systematization of LDP methods for descriptive statistics, including mean estimation, variance estimation, frequency and distribution estimation, contingency tables, range queries, and quantile estimation. We compare the methods in terms of their properties, requirements, and error bounds (where available) and show empirical comparisons for the most common applications. Our main contributions are as follows:

1. We provide a systematization of LDP methods for descriptive statistics, including mean estimation, variance estimation, quantile estimation, and distribution estimation. We compare the methods in terms of their properties, requirements, and error bounds.

2. We show that several mean estimation methods based on sampling from a Bernoulli distribution are equivalent in the one-dimensional case.

3. We generalize a method for variance estimation from mean estimation methods and provide an error bound for the variance estimate.

4. We empirically compare methods for mean estimation, variance estimation and frequency estimation and give recommendations for the choice of method.

5. We discuss the limitations of LDP methods for data analysis and open problems.

Related Privacy Definitions. While this work only focuses on pure (or approximate) local differential privacy, other variants of local differential privacy exist. One such variant is personalized local differential privacy, where each participant can choose a different privacy budget [AH17; SXY21; XZW21b]. Another variant, metric LDP, relaxes LDP's requirement of indistinguishability over the whole domain [Alv+18b]. This may allow an adversary to learn some approximate information about the private value while still hiding the exact value and enables a more accurate data analysis in many applications. This variant can be applied in all domains provided with a metric, and it provides a better privacy-utility trade-off especially with more sophisticated notions of statistical utility, such as those that measure the quality of a distribution estimation in terms of the earth mover's distance (aka Wasserstein distance or Kantorovich-Rubinstein metric) [Alv+18a; BP24; EP20]. For this reason, metric LDP is particularly successful in those applications that require an estimation of distributions (or frequencies, or histograms) that takes into account the ground distance. It has been applied in a wide range of domains, including location privacy [BP24], private text processing [FDM19], and private and personalized federated learning [Gal+23].

Organization. In Section 2, we overview key concepts and definitions related to LDP and descriptive statistics. Section 3 summarizes LDP methods for descriptive statistics. In Section 4, we present the results of our empirical comparison of LDP methods for mean, variance, and frequency estimation. We discuss the limitations and open topics in Section 5 before concluding in Section 6. In the appendix, Table 4 provides an overview of the notation and Tables 5 and 6 summarize all discussed algortihms.

2 Preliminaries

This section defines local differential privacy and related concepts, and provides an overview of descriptive statistics.

2.1 (Central) Differential Privacy

Dwork et al. [Dwo+06] introduced the concept of differential privacy, which provides a framework for quantifying the privacy loss resulting from computations on datasets containing sensitive information. The premise of differential privacy is that two nearly identical databases, differing by only one element, should yield similar outputs. An algorithm is defined as differentially private if its outputs for these databases fall within a specified closeness threshold. This guarantees that the outcome of the computation reveals little about an individual's data, thereby protecting privacy while still allowing the analysis of valuable aggregate data. More formally:

Definition 1 (Differential Privacy)**.** *A randomized algorithm $\mathcal{A}$ is* (ε, δ)-differentially private *if for all data sets D_1 and D_2 differing on at most one element, and all $\mathcal{S} \subseteq$ Range($\mathcal{A}$):*

$$\Pr[\mathcal{A}(D_1) \in \mathcal{S}] \leq \exp(\varepsilon) \Pr[\mathcal{A}(D_2) \in \mathcal{S}] + \delta,$$

If $\delta = 0$, we say that $\mathcal{A}$ achieves ε-DP.

Achieving differential privacy involves the injection of a precise amount of random noise into the algorithm's output, effectively obscuring the impact of any individual data point. This noising approach strikes a balance between preserving privacy and maintaining the utility of the data, allowing insights to be gained without compromising individual privacy. One of the biggest advantages of using differential privacy is that the privacy guarantees are information-theoretical and hold against adversaries with unbounded computational power. This means no matter which computation is done *after* the data is published in a differentially private way, the privacy bounds still hold. Yet, central DP has a downside: It allows data to be disclosed in a privacy-compliant manner after it has been processed by the curator $\mathcal{A}$. However, all data must first be given to $\mathcal{A}$, which requires trusting the curator. To remove this trust assumption, local differential privacy was proposed.

2.2 Local Differential Privacy

In contrast to the central model of differential privacy, the local model operates without a trusted curator. The local model of differential privacy was first formalized by Kasiviswanathan et al. [Kas+11]. In the local model, each client randomizes its data before sending it to the server that aggregates and potentially publishes it. This minimizes the trust required. The algorithm performing the randomization on each client is often called *local randomizer*. An adversary observing the output of the local randomizer should not be able to infer the private input as any possible input value is similarly likely to have generated the observed output. More formally:

Definition 2 (Local Differential Privacy)**.** *A randomized algorithm $\mathcal{A}$ with domain D is* (ε, δ)-locally differentially private *(an (ε, δ)-DP local randomizer) if for all $\mathcal{S} \subseteq Range(\mathcal{A})$ and for all pairs of client's values $x, y \in D$:*

$$\Pr[\mathcal{A}(x) \in \mathcal{S}] \leq \exp(\varepsilon) \Pr[\mathcal{A}(y) \in \mathcal{S}] + \delta,$$

If $\delta = 0$, we say that $\mathcal{A}$ is an ε-DP local randomizer (or (purely) ε-LDP).

2.3 Sequential Composition

While local differential privacy ensures the privacy of individual queries, the question arises as to how we can integrate multiple queries without violating overall privacy. Sequential composition solves this dilemma by showing that the aggregate privacy loss from sequentially applying different differentially private algorithms is limited to the sum of their individual privacy losses. Sequential composition is a strength of (local) differential privacy since the obtained bounds hold without any special effort by the curator. We now recite the theorem of sequential composition for central differential privacy and refer the reader to [DR14] for the proof. It should be noted that the theorem also holds for local differential privacy when considering databases of size 1.

Theorem 1 (Sequential Composition [DR14])**.** *Let $\mathcal{A}_i$ be an $(\varepsilon_i, \delta_i)$-differentially private algorithm for $1 \leq i \leq \lambda$. Then $\mathcal{A} := (\mathcal{A}_1, \ldots, \mathcal{A}_\lambda)$ is $(\sum_{i=1}^{\lambda} \varepsilon_i, \sum_{i=1}^{\lambda} \delta_i)$-differentially private.*

2.4 Interactivity

Local differential privacy algorithms can be classified into three categories: non-interactive, sequentially interactive, and fully interactive. In the non-interactive setting, the server assigns a local randomizer to each client before the clients send their responses to the server. In the sequentially interactive setting, the server may query clients with adaptively chosen local randomizers based on the responses of previous clients, but may only query each client once. In the fully interactive setting, the server is permitted to query each client multiple times with adaptively chosen local randomizers (ensuring that the privacy guarantees remain intact throughout the interaction). The majority of the methods discussed in this paper are non-interactive, with the exception of a few sequentially interactive methods that are explicitly marked as such. We refer the reader to the work by Joseph et al. [Jos+19b] for a detailed discussion of interactivity in local differential privacy.

2.5 Descriptive Statistics

Descriptive statistics form the foundation of data analysis and are employed to analyze data prior to the application of other methods, such as inferential statistics or machine learning [Bla15; PP20]. Descriptive statistics describe or summarize the main features of a dataset, which can consist of quantitative or categorical data. Quantitative data can be continuous (e.g., body weight) or discrete (e.g., number of children). Categorical data consist of values from a finite set of categories and can be unordered (e.g., blood type) or ordered (e.g., cancer stage). For continuous data, measures of central tendency (mean, median) and variability (standard deviation/variance, range/min/max, interquartile range) can be employed. In the case of unordered categorical data, the absolute or relative frequencies of each category can be calculated (i.e., the number of occurrences of each category divided by the total number of observations). This allows for the creation of contingency tables, which summarize the relationship or joint distribution between two categorical variables (providing frequencies for each combination of categories).

3 Descriptive Statistics Under Local Differential Privacy

This section provides an overview of methods for estimating descriptive statistics under local differential privacy.

3.1 Mean Estimation

The most common statistic is the mean, which describes the central tendency of a data set. Formally, we are given a data set $X = \{x_1, \ldots, x_n\}$ with $x_i \in \mathbb{R}^d$ and wish to estimate the sample mean $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$. In certain instances, we are also interested in estimating the population mean μ (i.e., the mean of the underlying distribution $\mu = \mathbb{E}[\mathcal{P}]$, $x_i \sim \mathcal{P}$). While we are generally interested in estimating the mean of data in $\mathbb{R}^d$, we will see that most methods require the data to be bounded. While these methods require specific input ranges, data from many applications can be transformed into this range by scaling and

Table 1: Comparison of mean estimation mechanisms for bounded data. All methods are non-interactive and are purely ε-LDP.

Algorithm	**Input Range**	**Error**
Laplace Duchi, Jordan, and Wainwright [DJW14; DJW18]	$x_i \in [-1, 1]$	minimax squared error [DJW18]: $O\left(\frac{1}{n\varepsilon^2}\right)$ for $\varepsilon \in [0, 1]$:
- for ℓ_2 [DJW14; DJW18]	$x_i \in \mathbb{R}^d, \|x_i\|_2 \leq r$	- minimax squared error: $O\left(r^2 \frac{d}{n\varepsilon^2}\right)$
- for ℓ_∞ [DJW14][a]	$x_i \in \mathbb{R}^d, \|x_i\|_\infty \leq r$	- minimax ℓ_∞ error: $O\left(\frac{r\sqrt{d \log(2d)}}{\sqrt{n\varepsilon^2}}\right)$
- for 1-sparse ℓ_∞ [DJW14]	$x_i \in \mathbb{R}^d, \|x_i\|_\infty \leq r, \|x_i\|_0 = 1$	- minimax squared error: $O\left(r^2 \frac{d \log(2d)}{n\varepsilon^2}\right)$
Nguyên et al. [Ngu+16][a]	$x_i \in [-1, 1]^d$	with prob. $1 - \beta$: $\|\hat{\mu} - \bar{x}\|_\infty = O\left(\frac{\sqrt{d \log(d/\beta)}}{\varepsilon\sqrt{n}}\right)$
Ding, Kulkarni, and Yekhanin [DKY17][a] Wang et al. [Wan+19a]	$x_i \in [0, m]$	with prob. $1 - \beta$: $\|\hat{\mu} - \bar{x}\| \leq \frac{m}{\sqrt{2n}} \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \sqrt{\log(2/\beta)}$
- Piecewise Mechanism	$x_i \in [-1, 1]$	with prob. $1 - \beta$: $\|\hat{\mu} - \bar{x}\| = O\left(\frac{\sqrt{\log(1/\beta)}}{\varepsilon\sqrt{n}}\right)$
- Hybrid Mechanism	$x_i \in [-1, 1]^d$	with prob. $1 - \beta$: $\|\hat{\mu} - \bar{x}\|_\infty = O\left(\frac{\sqrt{d \log(d/\beta)}}{\varepsilon\sqrt{n}}\right)$
Waudby-Smith, Wu, and Ramdas [WWR23][a]	$x_i \in [0, 1]$	with prob. $1 - \beta$: $\|\hat{\mu} - \mu\| \leq \frac{1}{\sqrt{2n}} \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \sqrt{\log(1/\beta)}$

[a] These methods are equivalent for $d = 1$ (see Proposition 1).

shifting the data (e.g., by using the min-max scaling if the bounds of the data are known). Some methods only handle 1-dimensional data (i.e., a single scalar value per participant) while others specifically focus on d-dimensional data (i.e., a vector or multiple attributes per participant).

In the following, we summarize the mean estimation methods and split them into two categories: methods for bounded data and Gaussian data. For each algorithm, we provide a brief description in the text and give details such as the input range and the error of the algorithms in Table 1 for bounded data and Table 9 in the appendix for Gaussian data. We also briefly summarize special cases of mean estimation that may be of interest for specific applications but are not directly comparable to the other methods.

3.1.1 Mean Estimation for Bounded Data

In the 1-dimensional setting, Dwork et al. [Dwo+06] introduced the Laplace mechanism for central DP, which can also be used for LDP. The Laplace mechanism involves adding noise, drawn from a Laplace distribution, to each value $x_i \in [-1, 1]$. The mean of these noisy values is then calculated as $\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} (x_i + Lap(\frac{2}{\varepsilon}))$. Duchi, Jordan, and Wainwright [DJW14; DJW18] show that the Laplace mechanism is asymptotically optimal for $d = 1$. They further provide mechanisms for $d \geq 1$, either bounded by the ℓ_2 or ℓ_∞ norm, both of which are based on sampling from a Bernoulli distribution. The ℓ_∞ mechanism additionally randomly rounds each dimension of the input to $-r$ or r. They show that both mechanisms are unbiased (i.e., the expected value of the responses is the true mean) and provide minimax squared error bounds for both cases. In the earlier work by Duchi, Wainwright, and Jordan [DWJ13], they provide error bounds for the ℓ_∞-norm case for general data, while in the later work by Duchi, Jordan, and Wainwright [DJW14], they only provide error bounds for the ℓ_∞ mechanism for 1-sparse data, which is data where only one dimension is non-zero.

Nguyên et al. [Ngu+16] claim to find issues in the method by Duchi, Jordan, and Wainwright [DJW14] and Duchi, Wainwright, and Jordan [DWJ13] and aim to fix them with their method. At the core, their method only handles one dimension and samples

from a Bernoulli distribution with a probability that depends on the input value. In fact, their method is equivalent to the ℓ_∞ mechanism by Duchi, Jordan, and Wainwright [DJW14] for $d = 1$ (see Proposition 1 below). To enable the method to handle d-dimensional data, they randomly select one dimension for each user and only transmit the response for this dimension.

Ding, Kulkarni, and Yekhanin [DKY17] propose 1BitMean which estimates the mean of data in the range $[0, m]$. An input x_i is rounded to 1 with probability $\frac{x_i}{m}$ and 0 otherwise. The resulting bit is flipped with probability $\frac{1}{e^\varepsilon+1}$ (by sampling from a Bernoulli distribution) and transmitted to the aggregator. The aggregator corrects for the bit flipping to obtain an unbiased estimate for the mean.

Wang et al. [Wan+19a] (who mostly consist of the same authors as Nguyên et al. [Ngu+16]) handle multiple dimensions by randomly selecting $k \leq d$ dimensions for each user and only transmitting mechanism responses for these dimensions. The authors combine the 1-dimensional case of the ℓ_∞ mechanism by Duchi, Jordan, and Wainwright [DJW14] with the introduction of the Piecewise mechanism to create the Hybrid mechanism. The Piecewise mechanism randomly samples a value from a range $[-D, D]$ (where D depends on ε), where values close to the input have the same high probability of being sampled and values further away have the same low probability. The Hybrid mechanism randomly selects a mechanism to use based on ε – with a higher probability for the Piecewise mechanism for large ε and the ℓ_∞ mechanism for small ε.

Waudby-Smith, Wu, and Ramdas [WWR23] present methods for estimating the population mean and a corresponding confidence interval using a generalization of 1BitMean [DKY17]. When using the default parameters, the mean estimation part of their method reduces to 1BitMean.

We find that all Bernoulli-based methods [DJW14; Ngu+16; DKY17; WWR23] are equivalent for $d = 1$. We formalize this observation in the following proposition and give the proof in the appendix.

Proposition 1. *The Bernoulli-based mechanisms M_{Du} by Duchi, Jordan, and Wainwright [DJW14] (ℓ_∞ case), M_N by Nguyên et al. [Ngu+16], M_D by Ding, Kulkarni, and Yekhanin [DKY17], and M_W by Waudby-Smith, Wu, and Ramdas [WWR23] (with default parameters) are equivalent for $d = 1$, i.e., they sample the response from the same probability distribution given the same input (in the corresponding input range).*

3.1.2 1-Dimensional Mean Estimation for Gaussian Distributions

In addition to the methods discussed in the previous sections, it is worth noting that there are approaches specifically designed to estimate the mean of unbounded data. Specifically, Gaboardi, Rogers, and Sheffet [GRS19] and Joseph et al. [Jos+19a] provide methods for estimating the mean of a Gaussian distribution (see Table 9 in the appendix).

Gaboardi, Rogers, and Sheffet [GRS19] aim to estimate a confidence interval for the mean of an unknown Gaussian distribution. They assume that the population mean is bounded in $[-R, R]$ and provide variants for known and unknown variance. Both variants are (ε, δ)-LDP with $\delta > 0$. Furthermore, both variants are sequentially interactive and use multiple rounds of communication.

Joseph et al. [Jos+19a] provide a set of algorithms to estimate the mean of an unknown Gaussian distribution with known or unknown variance. Furthermore, their algorithms

are strictly ε-LDP ($\delta = 0$) and require at most 2 rounds of communication (i.e., they provide non-interactive and sequentially interactive variants).

3.1.3 Special Cases of Mean Estimation

Further special cases of mean estimation have been proposed in the literature. Bhowmick et al. [Bho+19] and Asi, Feldman, and Talwar [AFT22] introduce methods for transmitting data sampled from the unit sphere, which is specifically useful for applications in machine learning. They claim that their algorithms are optimal, but also relax the privacy setting compared to standard ε-LDP. Xue, Zhu, and Wang [XZW21b] provide algorithms for mean estimation with personalized LDP (i.e., every data point x_i is perturbed using a different ε_i). Mean estimation for key-value pairs has been discussed by Ye et al. [Ye+19] and Gu et al. [Gu+20]. The estimation of means of sparse vectors has been discussed by Zhou et al. [Zho+22] and Duchi, Jordan, and Wainwright [DJW18].

3.2 Standard Deviation & Variance

Next to the mean, the standard deviation and variance are probably the most ubiquitous statistics. Per its definition [Bla15] and following their notations, the unbiased sample variance s_X^2 can either be calculated by subtracting the mean from each value as $s_X^2 = \frac{1}{n-1}\sum_i (x_i - \mu_X)^2$ or directly from the mean and the mean of the squared values as $s_X^2 = \frac{n}{n-1}(\mu_{X^2} - \mu_X^2)$.

The first option can be implemented through sequential interactivity by first estimating the mean using a subset of the participants and then estimating the variance using the remaining participants and the estimated mean. Given an ε-LDP mean estimation method, the resulting variance estimation is also ε-LDP as each participant is only queried by an ε-LDP mechanism once. Note that this method is necessarily sequentially interactive and requires two rounds of communication as the mean estimate is a prerequisite for the variance estimate.

The second option can be implemented non-interactively by estimating the mean and the mean of the squared values simultaneously. Ding et al. [Din+18] discuss how their 1BitMean algorithm can be used to estimate the mean μ_X and the mean of the squared values μ_{X^2} and use those estimates to calculate the variance. By splitting the privacy budget between the two estimations (ε_1 and ε_2 for the mean of X and X^2 respectively), they can provide an estimate for the variance with a total privacy budget of $\varepsilon = \varepsilon_1 + \varepsilon_2$ (sequential composability). Similarly, Waudby-Smith, Wu, and Ramdas [WWR23] discuss the estimation of the population variance using their method for estimating confidence intervals for the mean.

We now generalize these insights to other mean estimation algorithms and provide an upper bound for the error of the sample variance estimate. Assume an ε-LDP mean estimation method $\hat{\mu}$ with error $|\hat{\mu} - \mu| \leq f(n, \varepsilon)$ (only depending on n and ε). Using this method with privacy budget ε_X and n_X participants, the mean $\mu_X = \frac{1}{n}\sum_i x_i$ can be estimated with error $|\hat{\mu}_X - \mu_X| \leq f(n_X, \varepsilon_X)$. Analogously, the mean of the squared values $\mu_{X^2} = \frac{1}{n}\sum_i x_i^2$ can be estimated with error $|\hat{\mu}_{X^2} - \mu_{X^2}| \leq f(n_{X^2}, \varepsilon_{X^2})$. The sample variance s_X^2 can then be estimated as $\hat{s}_X^2 = \frac{n}{n-1}(\hat{\mu}_{X^2} - \hat{\mu}_X^2)$. It is now possible to either split the participants into two groups and estimate both means with the full privacy

budget ($n = n_X + n_{X^2}$ and $\varepsilon = \varepsilon_X = \varepsilon_{X^2}$) or to split the privacy budget and include all participants in the estimation of both means ($n = n_X = n_{X^2}$ and $\varepsilon = \varepsilon_X + \varepsilon_{X^2}$). The first method is ε-LDP as it applies an ε-LDP mechanism to each participant once. The second method is ε-LDP by sequential composability if $\varepsilon = \varepsilon_X + \varepsilon_{X^2}$.

The error of the non-interactive sample variance estimate can be calculated as follows:

Proposition 2. *Given a dataset $X = \{x_1, x_2, \ldots, x_n\}, x_i \in [-1, 1]$ and an ε-LDP mean estimation method $\hat{\mu}$ with error $|\hat{\mu} - \mu| \leq f(n, \varepsilon)$, the non-interactive sample variance estimator described above has error*

$$|\hat{s}_X^2 - s_X^2| \leq \frac{n}{n-1} \left(f(n_{X^2}, \varepsilon_{X^2}) + f(n_X, \varepsilon_X)^2 + 2f(n_X, \varepsilon_X) \right). \tag{1}$$

This result allows the estimation of a private dataset's mean and variance in one step, as the mean estimate is a necessary prerequisite for the variance estimate. Note that the error of the variance estimate is always at least as large as the error of the mean estimate. By selecting appropriate values for ε_X and ε_{X^2}, n_x and n_{X^2}, the errors of the mean and the variance estimates can be balanced. An aggregator may allocate more privacy budget (or participants) to the mean estimate if the mean is more important for their application, or vice versa. The exact impact of the privacy budget allocation on the error of the variance estimate depends on the error function of the mean estimation method.

3.3 Frequency & Distribution Estimation

Another important task in data analysis is estimating the data distribution. We first discuss this in the form of frequency estimation of categorical values, before moving to the estimation of histograms and probability density functions for numerical values (often called "distribution estimation" in literature).

3.3.1 Frequency Estimation

In frequency estimation, we have n data owners, each owning a single categorical value x_i from a domain $\mathcal{D}$ of size $|\mathcal{D}| = k$.[1] For a given item $x \in \mathcal{D}$, we define the frequency[2] as $f(x) = \frac{1}{n}|\{i \in [n] \mid x_i = x\}|$. The goal of LDP frequency estimation is to privately obtain an estimate $\hat{f}$ of f (often called a frequency oracle). Note that in this paper, we are interested in the relative frequency, i.e., $\sum_{x \in \mathcal{D}} f(x) = 1$. Postprocessing may be necessary to ensure that the sum of frequencies equals 1 and to improve the accuracy of frequency estimates from the input data. Methods for this range from simple normalization (dividing the frequencies of a value by the sum of all frequencies) to more advanced techniques like the matrix inversion method by Kairouz, Bonawitz, and Ramage [KBR16]. A more sophisticated approach is the Iterative Bayesian Update (IBU) [EP20; ACP23; MHS18], which is a form of the expectation maximization (EM) method. IBU computes the maximum likelihood estimator (MLE) of input frequencies based on output frequencies. Additional postprocessing methods are discussed by Cormode, Maddock, and Maple

[1] In literature the domain size is sometimes denoted d instead of k. To avoid confusion with the dimensionality d of the data we use k for the domain size.

[2] Note that the frequency $f(x)$ is unrelated to the error function $f(n, \varepsilon)$ used in Section 3.2.

[CMM21]. A common approach used by some methods (and applied to all methods in this paper) is the projection onto the probability simplex. The probability simplex is defined as $\Delta_k = \{p \in \mathbb{R}^k \mid p \geq 0, \sum_{i=1}^{k} p_i = 1\}$ and represents the set of all (valid) probability distributions over k categories [DJW14]. The projection of a vector p onto the probability simplex is defined as $\Pi_{\Delta_k}(p) = \arg\min_{q \in \Delta_k} \|q - p\|^2$.

Literature also considers the problem of finding heavy hitters, which are often defined as items x with $f(x) \geq \phi$ for some threshold ϕ or the top-l items with the largest frequencies. We do not cover heavy hitters in this paper and refer the reader to Cormode, Maddock, and Maple [CMM21] for an overview and the relevant literature for details [EPK14; BS15b; Qin+16; Bas+20; WLJ21].

Randomized response is the basis for many frequency estimation methods and was first introduced by Warner [War65] for binary data. The original idea was to give survey participants plausible deniability for sensitive questions. It works by flipping a biased coin and answering truthfully with probability p and answering randomly with probability $1 - p$. Randomized response is ε-LDP if $p = \frac{e^\varepsilon}{e^\varepsilon + 1}$ [War65; Wan+17].

Frequency oracles are the main component of locally differentially private frequency estimation, but vary in their construction, accuracy and the size of domain they are best suited for. Wang et al. [Wan+17] unify a number of frequency oracles under their proposed concept of pure LDP protocols. Note that this "pure" differs from the "pure" in pure LDP protocols (see Section 2.2) and refers to the simplicity of the protocols. In the following definition and the rest of the paper, we use "pure" to refer to the type of the protocols introduced by Wang et al. [Wan+17] and not the LDP property. Pure LDP protocols rely on an additional function $\mathsf{Support}(z)$, which is selected by the mechanism designer. This function defines the set of items that a given output z should be mapped to during frequency estimation, and therefore also influences the perturbation step. Informally, $\mathsf{Support}(z)$ can be understood as representing the idea that observing output z "supports" the "hypothesis" that the true value lies within the set $\mathsf{Support}(z)$. We give examples for the definition of this function for some mechanisms in the following paragraphs.

Definition 3 (Pure LDP Protocol [Wan+17])**.** *A protocol* PE *is a pure LDP protocol if and only if there exists a function* $\mathsf{Support}$ *and two probability values* $p^* > q^*$ *such that for all* v_1*,*

$$Pr[\mathsf{PE}(v_1) \in \{z \mid v_1 \in \mathsf{Support}(z)\}] = p^*,$$
$$\forall v_2 \neq v_1 Pr[\mathsf{PE}(v_2) \in \{z \mid v_1 \in \mathsf{Support}(z)\}] = q^*$$

Pure protocols are ε-LDP if $p^*/q^* = e^\varepsilon$. Responses z_i for $i \in [n]$ from pure LDP protocols can be used to estimate the frequency of an item x as $\hat{f}(x) = \frac{1}{p^* - q^*}\left(\sum_j \mathbf{1}_{x \in \mathsf{Support}(z_i)} - nq^*\right)$.

We now briefly summarize pure and non-pure LDP frequency estimation protocols. The following protocols are known to be pure LDP protocols [Wan+17; CMM21] (see Cormode, Maddock, and Maple [CMM21] for a more detailed description):

Direct encoding or k-ary randomized response (k-RR; sometimes d-RR in literature) was first introduced by Kairouz, Oh, and Viswanath [KOV14; KOV16] and generalizes randomized response to k-ary data. The mechanism's output space is equal to the input

space and the probability of reporting the true value is $p = \frac{e^\varepsilon}{e^\varepsilon + k - 1}$, while the probability of reporting any other value is $q = \frac{1}{e^\varepsilon + k - 1}$. In this case, $\mathsf{Support}(z) = \{x \mid x = z\}$, i.e., we count every response as though it were the true value.

Another method, proposed by Wang et al. [Wan+17], is histogram encoding. It works by encoding the input into a "histogram" B, i.e., a vector of size k with a 1 indicating the index of the item and 0's elsewhere. Each participant now perturbs each entry of this vector with Laplace noise ($B'[i] = B[i] + Lap(\varepsilon/2)$) and sends the perturbed vector B' to the aggregator. Reports can be aggregated using summation (SHE), where all noisy reports are summed up in a noisy frequency estimate. SHE is not a pure protocol as there is no known Support function. Alternatively, thresholding (THE) can be used, where each noisy vector B' is interpreted as a binary vector through the definition of $\mathsf{Support}(B') = \{i \mid B'[i] > \theta\}$. The intuition here is that θ is used to distinguish between samples from the two possibly overlapping distributions $1 + Lap(\varepsilon/2)$ and $0 + Lap(\varepsilon/2)$. The binarized vectors are then summed up to result in a frequency estimate. Wang et al. [Wan+17] claim that an optimal θ can be found numerically to be in $(\frac{1}{2}, 1)$ and depends on ε.

Unary encoding methods encode the input into a one-hot-encoded binary string (a vector of size k with a 1 indicating the index of the item and 0's elsewhere) and independently flip the single 1 bit with probability $1 - p$ and the 0 bits with probability q. To the best of our knowledge, this method was first introduced by Duchi, Wainwright, and Jordan [DWJ13]. Symmetric unary encoding (SUE) uses $p + q = 1$ (equivalent to the basic RAPPOR protocol [EPK14]). Optimized unary encoding (OUE) was introduced by Wang et al. [Wan+17] and uses optimized choices for p and q. In both cases, $\mathsf{Support}(B) = \{i \mid B[i] = 1\}$, i.e., we "decode" the one-hot encoding and count every response as though it were the true value.

Following the ideas of RAPPOR (see non-pure protocols below), Wang et al. [Wan+17] propose local hashing methods. In the local hashing approach, users randomly pick a hash function H to map the input to a smaller output space of size g and then apply direct encoding to the output of the hashed values. Binary local hashing (BLH) uses binary hash functions with $g = 2$ and optimal local hashing (OLH) uses hash functions with $g = e^\varepsilon + 1$. Here, $\mathsf{Support}(\langle H, z \rangle) = \{x \mid H(x) = z\}$, i.e., the set of items that are hashed to the observed output y. Bassily and Smith [BS15a] use random matrix projection to reduce the dimensionality of the data to one bit and then use randomized response to perturb the bit, which is logically equivalent to BLH according to Wang et al. [Wan+17]. Fast local hashing (FLH) [CMM21] proposes a heuristic modification to speed up OLH by reducing the number of hash functions to sample from.

Hadamard methods are based on the Hadamard transform which is closely related to the discrete Fourier transform. The Hadamard mechanism (HM) [Bas+20] samples an index j and calculates the corresponding Hadamard coefficient of the input vector and the index j. It reports this coefficient using direct encoding. This allows the aggregator to estimate the Hadamard coefficients and use them to approximate the frequency of any item. The related Hadamard response protocol (HR) [ASZ19] also reports a random Hadamard coefficient, but does not randomly choose a fixed index. Instead, it randomly chooses whether to report a positive or negative coefficient and chooses an appropriate index j.

Wang et al. [Wan+17] give some guidelines for the selection of an appropriate pure

protocol: For small domains ($k < 3e^{\varepsilon} + 2$), direct encoding is the best choice, whereas OUE should be used for larger domains if its communication cost is acceptable. When the domain is so large that the communication cost is a concern, OLH is the best choice. Cormode, Maddock, and Maple [CMM21] add that FLH is several times faster than OLH with comparable accuracy and that Hadamard-based methods are orders of magnitudes faster and almost as accurate in extremely large domains (more than thousands of items).

Next to pure protocols, there are also protocols that do not fit the definition of pure protocols (or where no mapping to pure protocols is known). Erlingsson, Pihur, and Korolova [EPK14] introduced the well-known RAPPOR protocol, which uses a Bloom filter to project the input to a smaller space of fixed size and then applies randomized response. Its basic version is pure (see above) and similar to earlier work by Duchi, Wainwright, and Jordan [DWJ13].

Kairouz, Bonawitz, and Ramage [KBR16] show that k-RR and basic RAPPOR are order-optimal for frequency estimation in certain privacy regimes (k-RR in the regime where $\varepsilon \approx \log(k)$ and basic RAPPOR in the regime where ε is close to 0). They introduce the O-RR mechanism based on k-RR and hash functions for settings with domains that are not enumerable, but show that this method also outperforms k-RR and RAPPOR on closed domains of size k when using permutations instead of hash functions. The downside to this method is that the aggregator needs to choose the number of cohorts that each use a different hash function or permutation in advance and the authors do not discuss how to choose this number optimally.

Wang et al. [Wan+16] and Ye and Barg [YB17] independently propose the l-Subset mechanism (k-Subset in literature), in which each participant reports a random subset of size $l < k$ of the input domain, which contains the true value with a certain probability. Ye and Barg [YB17] claim that this method fills the gap between RAPPOR and k-RR and is optimal for "medium" privacy regimes that are far from 0 and $\log(k)$. They show that their method can improve utility when $3.8 \ll \varepsilon \ll ln(l/9)$.

Nguyên et al. [Ngu+16] base their frequency estimation method on the work by Bassily and Smith [BS15a]. Instead of constructing a random matrix, they create a binary $k \times k$ matrix where any two column vectors are orthogonal. By letting each user randomly select one attribute to report, they enable a combination of frequency estimation and mean estimation for multiple categorical and numeric attributes without reducing the privacy budget of individual reports.

Murakami, Hino, and Sakuma [MHS18] specifically care about the setting where n or ε are small and propose a solution based on the IBU to improve the accuracy of the frequency estimation. They use k-RR to obtain noisy reports and then apply the IBU to reduce the estimation error of the frequencies.

ElSalamouny and Palamidessi [EP20] make two main contributions: First, they generalize the IBU postprocessing to the case of personalized privacy, and second, they compare it to other standard postprocessing methods used in LDP, in particular the matrix inversion method by Kairouz, Bonawitz, and Ramage [KBR16]. They show that while the IBU is equivalent to the matrix inversion method in the case of the k-RR mechanism, it outperforms it when applied to other obfuscation mechanisms such as those used in metric privacy. In general, IBU is the only known postprocessing method that is universally optimal, as it is shown to produce a maximum likelihood estimator regardless of the mechanism used for obfuscation.

3.3.2 Histogram Estimation

At the intersection of frequency and distribution estimation we find the estimation of histograms. A histogram is a discretization of the continuous data space into bins and the estimation of the frequency of data points in each bin. Note that the histogram estimation problem is a special case of the frequency estimation problem where each bin is a different item in the domain. For this reason, few methods are designed explicitly for histogram estimation.

Duchi, Wainwright, and Jordan [DWJ13] discuss histogram estimation as an approximation of the density estimation problem. They split the data space $[0, 1]$ into k equal-sized bins and replace each data point x_i with a one-hot vector of length k where the j-th entry is 1 if x_i falls into the j-th bin and 0 otherwise. Each vector is then perturbed using the Laplace mechanism. The aggregator then sums up the perturbed vectors to obtain counts for each bin. The counts are normalized and projected onto the k-dimensional probability simplex to obtain a differentially private estimate for the density. They note that the histogram estimator is also asymptotically optimal for the density estimation problem for Lipschitz densities.

Ding, Kulkarni, and Yekhanin [DKY17] introduce dBitFlip, a method for estimating histograms with k buckets. Their method works by sampling l bucket indices from $[k]$ for each user and responding with one bit for each selected bucket. The bits are drawn from a Binomial distribution with probability $e^{\varepsilon/2}/(e^{\varepsilon/2}+1)$ for the correct bucket and $1/(e^{\varepsilon/2}+1)$ for all other buckets. The aggregator sums up the received bits for each bucket and uses the noisy counts to estimate the histogram (compensating for the random bit flipping).

IBU postprocessing has been used for histogram estimation by Agrawal and Aggarwal [AA01]. Although their work predates the development of differential privacy and applies IBU to a different obfuscation mechanism, the method they proposed is general and can be applied to LDP as well.

3.3.3 Distribution Estimation

In distribution estimation, the goal is to estimate the probability density function of continuous data. Duchi, Jordan, and Wainwright [DJW18] argue that the Laplace mechanism does not provide optimal error bounds for distribution estimation. They then discuss minimax bounds for LDP density estimation for cases where the underlying density belongs to a Sobolev class defined using trigonometric functions as basis functions. For densities in the Sobolev class of order 1, histogram estimators (like the estimator by Duchi, Wainwright, and Jordan [DWJ13] discussed in the previous section) are asymptotically optimal. For densities with higher orders of smoothness (i.e., density functions that are more often differentiable), they develop an estimator based on orthogonal series expansions and show that it is asymptotically optimal.

Diao et al. [Dia+20] aim to model the data distribution as a Gaussian Mixture Model (GMM) and provide a method to estimate the parameters of the GMM in a differentially private manner. They build on the Gaussian mechanism and therefore only provide (ε, δ)-LDP instead of ε-LDP.

Li et al. [Li+20] introduce the square wave mechanism which is conceptually very similar to the Piecewise mechanism by Wang et al. [Wan+19a]. They construct a histogram

Table 2: Comparison of methods for estimating contingency and marginal tables under local differential privacy.

Method	# Attributes	Goal	Main Component
Fanti, Pihur, and Erlingsson [FPE16]	2 categorical	full contingency table	Expectation Maximization
Ren et al. [Ren+18]	d categorical	fixed k-way marginal	Expectation Maximization / Lasso regression
Cormode, Kulkarni, and Srivastava [CKS18]	d binary	all k-way marginals	Hadamard Transform on private data
Zhang et al. [Zha+18]	d categorical	all k-way marginals	Entropy Maximization + Frequency Oracle for sample marginal
Xue, Zhu, and Wang [XZW21a]	2 categorical	joint distribution	Extension of l-Subset for frequency estimation [Wan+16; YB17]

of the perturbed values and use an Expectation Maximization algorithm to estimate the underlying input distribution. They use the prior knowledge that the frequencies of neighboring numerical values are similar to introduce a smoothing step in the EM algorithm. The authors relate the core of their algorithm to frequency oracles (for numerical values) and show how their method can be used to estimate the mean, variance and quantiles of the input distribution. Their method works best for smooth input domains and is less effective for spiky distributions.

3.4 Contingency Tables & Marginal Tables

The methods in the previous section are designed for estimating distributions over univariate data, but many applications require the estimation of joint distributions over multiple attributes. In this section, we discuss methods for estimating contingency tables and marginal tables under local differential privacy. All methods in this section assume that there are up to d categorical attributes and that the domain of each attribute is known (see Table 2 for details). First, we need to define some terms: A full contingency table gives the joint distribution of all d attributes in a dataset, but may be very large and computationally expensive to estimate. The k-way marginal over a set A of $k < d$ attributes gives the joint distribution of the attributes in A. The set of all k-way marginals contains the marginals for all possible subsets of size k.[3]

Fanti, Pihur, and Erlingsson [FPE16] show how reports from RAPPOR can be used to estimate the joint distribution of two categorical variables (contingency table) by applying an expectation maximization algorithm to the noisy reports. They explain that their proposed method is not RAPPOR-specific and can be used with other locally differentially private encoding methods. The main limitation of their method is that it only works for two categorical variables and is not directly applicable to higher-dimensional data.

Ren et al. [Ren+18] aim to privately publish synthetically generated data with a similar joint distribution as the underlying discrete d-dimensional private data. Inspired by the work of Fanti, Pihur, and Erlingsson [FPE16], they use the EM algorithm to estimate the joint distribution of the private data, but introduce LASSO regression to deal with the sparsity of high-dimensional data. They then perform dimensionality reduction on the learned distribution and sample from the resulting low-dimensional distributions to synthesize an approximate dataset. Note that their method for estimating the joint distribution only works for a specific k-way marginal and does not provide a method for estimating the full contingency table or the set of all k-way marginals.

Cormode, Kulkarni, and Srivastava [CKS18] provide a method for estimating k-way

[3]Note that the $k < d$ in this section denotes a subset of attributes/dimensions and is different from the k in frequency estimation, which denotes the size of the domain.

marginals of a high-dimensional binary dataset under local differential privacy based on the discrete Fourier transform (Hadamard transform) which has previously been studied in the central differential privacy setting. The authors claim that in practical applications, analysts are often interested in lower-dimensional marginals and therefore provide a method of estimating all k-way marginals without having to select a specific subset of attributes in advance. They compare several algorithm variants and conclude that computing the Hadamard transform on the private data and releasing a random coefficient via randomized response is the most effective method for reconstructing any k-way marginal through postprocessing. They also compare their methods to the EM-based approach by Fanti, Pihur, and Erlingsson [FPE16] and claim that the EM-based approach does not provide any worst case guarantees for the accuracy and may terminate with a poor estimate.

Zhang et al. [Zha+18] propose the CALM method to estimate any k-way marginal of a high-dimensional dataset under local differential privacy. They claim that previous methods [FPE16; Ren+18; CKS18] are not practical for high-dimensional data. CALM is based on PriView, which was used in the central differential privacy setting, and works by first selecting m marginals of size l and then assigning each user to one of the selected marginals. The aggregator uses a frequency oracle to estimate the frequencies for the selected marginals. Since the selected marginals do not cover all possible k-way marginals, the authors rely on the postprocessing steps of PriView to estimate the remaining marginals: The resulting frequencies are first checked for consistency and non-negativity and then used to estimate the remaining marginals using maximum entropy estimation. The authors discuss that their method has three sources of error: noise errors from the frequency oracle, reconstruction errors when a k-way marginal is not covered by the selected marginals, and sampling errors since each marginal is only estimated from a subset of the population. Since the choice of m and l is crucial for the accuracy of the method and the reconstruction error depends on the dataset, the authors provide a method to determine these parameters based on some required error threshold.

Xue, Zhu, and Wang [XZW21a] propose the JESS method for estimating the joint distribution of two categorical attributes under local differential privacy. The method is inspired by the k-subset mechanism [Wan+16; YB17] for frequency estimation and extends it to transmitting two categorical attributes.

3.5 Range Queries

In this section, we discuss methods for estimating range queries on discrete ordinal data, which can be used to estimate quantiles and other statistics. Assuming n participants, each with a private value $x_i \in [k]$, a range query $R_{[a,b]} \geq 0$ with $a, b \in [k]$ and $a < b$ counts the relative frequency of participants with a value in the range $[a, b]$: $R_{[a,b]} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}_{a \leq x_i \leq b}$. The goal is to privately collect enough information upfront to be able to estimate any $R_{[a,b]}$ with a small error.

Cormode, Kulkarni, and Srivastava [CKS19] are the first to introduce range queries in the LDP setting, and discuss methods for performing range queries on discrete (ordered) one-dimensional data. They first consider the naive solution of summing up point queries (i.e., frequencies obtained through frequency oracles; see section 3.3) for each value in the range, but show that the variance of this approach grows linearly with the size of the

range. Inspired by methods from the central DP setting, they propose a method based on hierarchical histograms, where the variance of the estimate only grows logarithmically with the size of the range. Additionally, they provide a method based on the discrete Haar transform which has a similar variance growth, but empirically shows better results for small privacy budgets.

Wang et al. [Wan+19b] introduce the problem of answering multi-dimensional analytical queries, where the goal is to aggregate a non-private measure for participants for which certain constraints on their private data (point constraints for categorical data and range constraints for ordinal data) are satisfied. Their special case of COUNT-queries is comparable to range queries on multi-dimensional data of mixed types. They introduce HIO, which – for the one-dimensional case – works by building a b-way tree of height h, where each node represents an interval with b equally sized subintervals as children. A range query can then be answered by summing up the frequencies for the appropriate subintervals. Similar to the other methods, they use one frequency oracle (OLH [Wan+17]) per tree level and split the participants over the levels. Their base approach is extended to d ordinal dimensions by constructing multiple trees and taking the Cartesian product, resulting in $(h+1)^d$ d-dimensional tree levels. Each participant then responds their interval membership for one of these multi-dimensional levels using a frequency oracle. They additionally introduce categorical dimensions by constructing a tree of height 2, where the root covers the whole domain and has children for each possible value of the categorical dimension. Since their method does not scale well to large d, they propose a conjunctive frequency estimator which combines single-dimension responses from OLH to estimate the joint frequencies for multiple dimensions. However, in their experiments they only test the methods on settings with small $d \leq 4$ and the conjunctive method does not work well for multiple ordinal dimensions.

Li et al. [Li+20] show how their work on estimating the distribution of numerical one-dimensional data can be used to answer range queries. They compare against the method by Cormode, Kulkarni, and Srivastava [CKS19] on a number of real-world datasets and show that their method has lower error in most cases.

Yang et al. [Yan+20] estimate multi-dimensional range queries with what they call the hybrid-dimensional grid (HDG) approach. Their approach constructs grids for each individual dimension (similar to binning in histograms) and for all pairs of dimensions. Participants are split over the $d + \binom{d}{2}$ grids and asked to respond their cell membership using OLH [Wan+17]. These responses are then used to construct answers for λ-dimensional ($\lambda \leq d$) range queries by selecting relevant grids and performing maximum entropy estimation to combine the estimates. This approach is very similar to CALM [Zha+18] for marginal estimation, which first collects l-way marginals ($l < k$) and uses maximum entropy estimation to estimate the requested k-way marginals.

Du et al. [Du+21] notice that sparse areas in the data space can lead to large errors for the respective sub-intervals or grid cells in the previous methods. They therefore propose AHEAD, which adaptively builds a hierarchical grid structure that avoids sparse regions (only cells/intervals with a large enough estimated frequency are further sub-divided). At each level of the hierarchy, they estimate the frequencies of the cells/intervals using OUE [Wan+17]. They perform experiments on real-world datasets and show that AHEAD outperforms the previous methods in terms of mean squared error.

3.6 Order Statistics: Quantiles, Median, Maximum, Minimum

As the final type of descriptive statistics, we now discuss order statistics, such as the median, quantiles, maximum, and minimum. Although they are rather important in non-private data analysis, we have waited until now to discuss these statistics because they are closely related or build on the methods we have discussed in the previous sections. Quantiles divide the data into equal-sized groups, with the median dividing the data into two equal-sized groups. The maximum and minimum represent the largest and smallest values in the data, respectively.

In local differential privacy, binary search has been used to estimate the median and quantiles. Cyphers and Veeramachaneni [CV17] first introduce the concept of using binary search for estimating the median in the LDP setting, but do not analyze the error of their method. Later, Gaboardi, Rogers, and Sheffet [GRS19] provide a method for estimating quantiles as part of their method for estimating the mean and variance of Gaussian distributions. For a given privacy budget and maximum deviation from the target quantile, they provide the required number of participants and search rounds to achieve this deviation with high probability. Finally, Fukuchi, Yu, and Sakuma [FYS22] aim to find the minimum (or maximum) value in a numerical 1-dimensional dataset bounded in $[-1, 1]$ using binary search in combination with randomized response and a fixed search depth.[4] While the approach is very straightforward, they show that the minimum finding problem is fundamentally difficult in the LDP setting and that no LDP mechanism can consistently estimate the minimum value under the worst case data distribution. They show that the problem is easier if the data has a larger minimum-side fatness, i.e., if more data points are close to the minimum. All three methods [CV17; GRS19; FYS22] require some initial bounded search interval that contains the target quantile and a search depth to be set in advance.

Another approach to estimating the median is to use stochastic gradient descent. Duchi, Jordan, and Wainwright [DJW18] provide a sequentially interactive method for estimating the median of a numerical 1-dimensional dataset in the LDP setting using stochastic gradient descent. The method starts with a random estimate for the median and sequentially asks each participant to provide a noisy answer to whether their value is larger or smaller than the current estimate. After each step, the estimate is updated based on the noisy answer and a decreasing learning rate. This procedure requires each participant to only respond once and therefore does not need to split the privacy budget between multiple rounds of communication (as the binary search methods do). The authors only discuss the method for $\varepsilon \leq 1$, and it is not immediately clear whether the method can be adapted to larger privacy budgets and other quantiles.

Next to binary search and gradient descent, range queries of the form $R_{[0,q]}$ (so-called prefix queries) can be used to estimate quantiles by setting q to the desired quantile [CKS19].

Table 3: Datasets used for the empirical evaluation. Type: Num=Numeric, Cat=Categorical. Domain: k is the number of categories for categorical data. Size: Number of data points.

Dataset	Type	Domain	Size	Notes
		Synthetic Data		
Uniform small	Num	$[0, 1]$	any	Uniform Distribution
Uniform large	Num	$[-100, 100]$	any	Uniform Distribution
Bimodal	Num	$[0, 1]$	any	$\mathcal{N}(0.3, 0.1) + \mathcal{N}(0.6, 0.2)$
Binomial	Num	$[0, 100]$	any	$\mathcal{B}(100, 0.2)$
Binomial	Cat	$k = \{8, 128\}$	any	$\mathcal{B}(k, 0.2)$
Geometric	Cat	$k = \{8, 16, \ldots, 512\}$	any	Geometric Distribution over k elements with $p = 5/k$, as in Kairouz, Bonawitz, and Ramage [KBR16]
		Real Data		
Adult	Num	$[16, 100]$	48 842	"Age" column from the UCI Adult dataset [BK96]
NYC Taxi	Num	$[0, 86400]$	8 760 687	"Pick-up time" column (in seconds) from the Yellow Taxi Trip records dataset for January 2018 [Cit02]
US Census	Cat	$k = 400$	2 458 285	US Census [MTH90] dataset processed to 400 binary attributes as in [MHS18]

4 Empirical Comparison

The previous section outlines many methods estimating various descriptive statistics under local differential privacy. However, it is not immediately clear which method is best suited for a given task, number of participants, or privacy budget. In this section, we aim to provide an empirical comparison of the methods for estimating the mean and the variance of numerical data and estimating the frequency of categorical data. We do not provide comparisons for the other statistics, as the relevant methods often have differing goals or requirements and are not directly comparable.

For the empirical comparisons, we used a number of synthetic and real-world datasets, which are summarized in Table 3 and visualized in Figure 17 in the appendix. For each numerical dataset, we defined a specific data range $[a, b]$, which was used to transform the data to the range required by the methods.

4.1 Mean Estimation

For the mean estimation task, we simulated each method from Table 1 with 100 different random seeds for each combination of n, ε, and dataset to account for the random nature of the methods. As many methods are designed for a specific input range, we first transformed the data to the required range, applied the method, and then transformed the result back to the original range. To evaluate the utility, we consider the mean squared error ($\frac{1}{n}\sum(\hat{\mu} - \bar{x})^2$) and the mean absolute error ($\frac{1}{n}\sum|\hat{\mu} - \bar{x}|$). To account for the different dataset and input ranges, we mainly consider "range-scaled" errors, where

[4]The method can potentially also be adapted to finding any quantile. However, the utility analysis in the paper only applies to finding the minimum/maximum as it is based on the fatness of the data near the minimum/maximum.

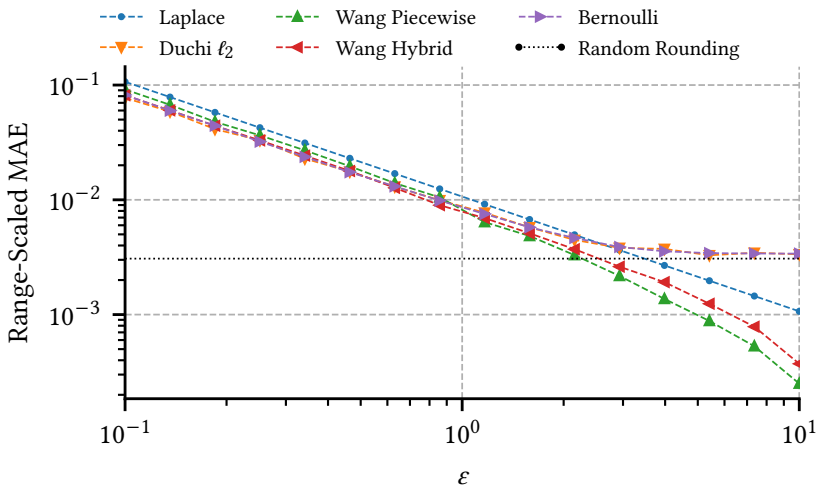

Figure 1: Mean absolute error of the mean estimation averaged over all datasets (scaled by the respective input range) and $n = 10^4$. "Duchi" and "Wang" refer to the works by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18] and Wang et al. [Wan+19a] respectively. Ding, Kulkarni, and Yekhanin [DKY17], Nguyên et al. [Ngu+16], and Waudby-Smith, Wu, and Ramdas [WWR23], and the ℓ_∞ mechanism by Duchi, Wainwright, and Jordan [DWJ13] are equivalent and are summarized as "Bernoulli Mechanisms". For better readability, we have omitted the standard deviations of the errors, which are shown in Figure 5 in the appendix. Figures 7, 8 and 9 show the mean absolute error, the mean squared error and response variances for the different methods, respectively.

the error is divided by the size of the input data range to allow for a comparison between datasets.

4.1.1 One-Dimensional Mean Estimation

Figure 1 shows the mean absolute error of the mean estimation methods for all numerical datasets (scaled by the respective input range) and $n = 10^4$. We observe that the error decreases with increasing privacy budget, but that the rate of decrease differs between the methods. While all methods show a similar error for $\varepsilon \lessapprox 1$, the errors deviate for larger ε, with Bernoulli methods [DWJ13; Ngu+16; DKY17; WWR23] showing the largest error and the methods by Wang et al. [Wan+19b] showing the smallest error. Looking more closely at the Piecewise and Hybrid methods by Wang et al. [Wan+19b], we see that the Hybrid method shows a lower error than the Piecewise method for small ε. However, for larger ε, the Piecewise method shows a lower error than the Hybrid method, although Wang et al. [Wan+19b] have constructed the Hybrid method to be optimal for all ε.

Further, we find that the Bernoulli mechanisms [DWJ13; DKY17; Ngu+16; WWR23] and the ℓ_2 mechanism by Duchi, Wainwright, and Jordan [DWJ13] show similar error rates and converge to the same (high) error for increasing ε. As ε increases, these methods reduce to randomly rounding the data, i.e., $Pr(z_i = 1) = x_i$ and $Pr(z_i = 0) = 1 - x_i$. We have also simulated this "random rounding" method and show the result in the relevant figures. We see that the error of the Bernoulli mechanisms indeed converge towards the

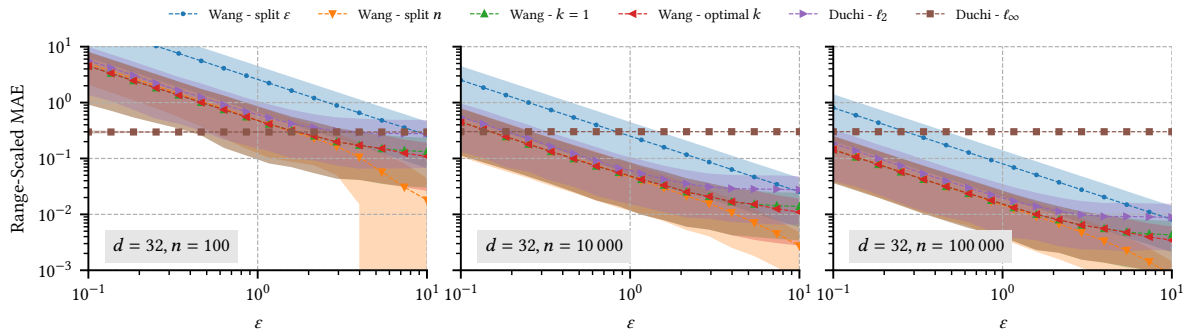

Figure 2: Mean absolute error of the multi-dimensional mean estimation (scaled by the input range) for $n = \{10^2, 10^4, 10^5\}$ and $d = 32$ for the Binomial dataset stacked d times. The methods refer to Wang et al. [Wan+19b] and Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18]. The "split" variants perform 1-dimensional mean estimation on each dimension, where split n splits the participants by the number of dimensions and split ε splits the privacy budget by the number of dimensions. The shaded areas show the standard deviation of the errors. See Figures 10 and 11 in the appendix for different combinations of n and d.

error of this random rounding procedure.

From Figures 7 and 8 in the appendix we see that the mean absolute error and the mean squared error behave similarly. Figure 9 in the appendix shows the variance of the responses for the different methods. Here we see that, similar to the errors, the variance of responses for all unbiased methods decreases with increasing privacy budget, but the decrease levels off as ε approaches 10. We also note that the variance of the method by Ding, Kulkarni, and Yekhanin [DKY17] is constant for all privacy budgets, which is due to the biased nature of the method (i.e., the mean of the noisy responses is not the true mean and needs correction to be applied).

4.1.2 Multi-Dimensional Mean Estimation

We now consider the multi-dimensional mean estimation task. For the evaluation, we use the Binomial dataset stacked d times to create a d-dimensional dataset. In addition to the ℓ_2 and ℓ_∞ methods by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18] and the method by Wang et al. [Wan+19b], we also add variants of the latter: First, instead of setting k optimally, we set $k = 1$ (as in the previous version of the paper [Ngu+16]). Additionally, we test the naive approaches of splitting the privacy budget or the participants by the number of dimensions.

Figure 2 shows the mean absolute error of the multi-dimensional mean estimation (scaled by the input range) for $n = \{10^2, 10^4, 10^5\}$ and $d = 32$ (see Figures 10 and 11 in the appendix for further combinations of n and d). We observe that the ℓ_∞ method [DWJ13] shows a constant error for all settings with $d > 1$. While Duchi, Wainwright, and Jordan [DWJ13] discussed this method for general mean estimation in the earlier version of their paper, the later publication [DJW18] only discuss its use for 1-sparse data, which could explain the bad performance for our dense data.

Out of the other methods, splitting the privacy budget by the number of dimensions

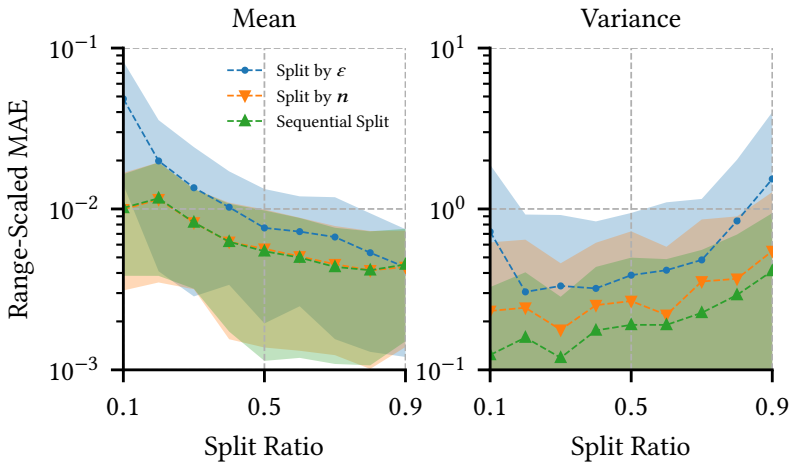

Figure 3: Mean absolute error of the variance estimation averaged over all datasets (scaled by the respective input range) with $n = 10^4$ and $\varepsilon = 2$. The split ratio defines how much of the privacy budget (or participants) is used for the mean estimation step. The variance estimation uses the Piecewise mechanism by Wang et al. [Wan+19a] as the underlying mean estimator. The shaded areas show the standard deviation of the errors. See Figures 12 and 13 in the appendix for different n and ε.

shows the highest error, whereas splitting the participants by the number of dimensions shows the lowest error. The other methods show a similar error, with the ℓ_2 method showing a slightly higher error than the method by Wang et al. [Wan+19b] (both for optimal k and $k = 1$). Interestingly, there is no substantial difference between setting k optimally and setting $k = 1$ for the method by Wang et al. [Wan+19b]. In fact, both variants are equivalent for $\varepsilon < 5$ as the optimal value $k = \max(1, \min(d, \lfloor \frac{\varepsilon}{2.5} \rfloor))$ is equal to 1 for $\varepsilon < 5$. The optimal variant only slowly increases k from 2 to 4 for ε between 5 and 10.

4.2 Variance Estimation

We have introduced and discussed three options for variance estimation in Section 3.2 and now evaluate their performance. We simulated each variant of variance estimation 20 times for each combination of n (between 10^2 and 10^7), ε (between 0.1 and 10), and dataset. In all cases, we used the Piecewise mechanism by Wang et al. [Wan+19a] as the underlying mean estimator. Figure 3 shows the range-scaled MAE of the estimated mean and variance over different split ratios (i.e., how much n or ε is used for the mean estimation step) for $n = 10^4$ and $\varepsilon = 2$. The split ratio defines how much of the privacy budget (or participants) is used for the mean estimation step. If the split ratio is 0.9, 90% of the privacy budget (or participants) is used for the mean estimation step, and if the split ratio is 0.5, half of the privacy budget (or participants) is used for the mean estimation step. See Figures 12 and 13 in the appendix for different n and ε. The error for the split by ε method is, on average, larger than that of the other two methods for both mean and variance estimation. This aligns with the general idea that splitting n is preferred over splitting ε when performing multiple queries in LDP [Wan+17].

The errors for the split by n and sequential split methods are similar for the mean

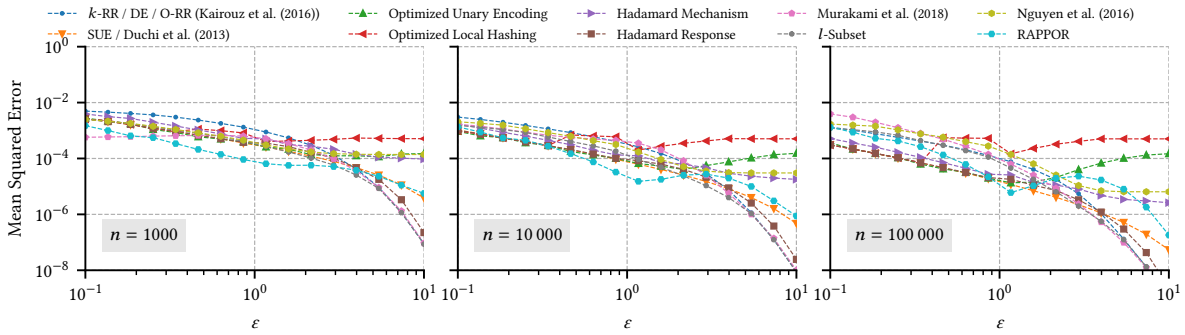

Figure 4: Mean squared error of the frequency estimation averaged over all datasets with $n = 10^3, n = 10^4, n = 10^5$. For better readability, we have omitted the standard deviations of the errors, which are shown in Figure 6 in the appendix.

estimation, which is expected as those methods are equivalent for the mean estimation. In the variance estimation, the sequential split method has a slightly lower error on average than the split by n method. However, since both errors show a large standard deviation, the difference is not significant (in fact, the split by n method shows a lower error for some individual simulations).

4.3 Frequency Estimation

We simulated all non-pure frequency oracles from Section 3.3 (RAPPOR [EPK14], l-Subset [YB17; Wan+16], and the methods by Kairouz, Bonawitz, and Ramage [KBR16] and Nguyên et al. [Ngu+16], and Murakami, Hino, and Sakuma [MHS18]) and the pure frequency oracles recommended by Wang et al. [Wan+17] and Cormode, Maddock, and Maple [CMM21] (i.e., k-RR/direct encoding, symmetric and optimized unary encoding, optimized local hashing, Hadamard mechanism, and Hadamard response) for different n, ε, and datasets of different domain sizes. Each simulation was repeated 20 times for each combination of n, ε, and dataset.

To ensure a fair comparison, we applied projection onto the probability simplex for all methods (implemented according to the algorithm by Wang and Carreira-Perpiñán [WC13]). Following literature, we used the mean squared error $\left(\frac{1}{d}\sum_{x\in D}(\hat{f}(x) - f(x))^2\right)$ and the variance of the frequency estimates as evaluation metrics. We calculate the variance as the mean over the sample variances of the individual frequency estimates.

We evaluated the frequency estimation methods on the categorical datasets from Table 3, where the Geometric distribution dataset was used with varying domain size to evaluate the impact of the domain size on the frequency estimation methods (see Figure 16 in the appendix).

Figure 4 shows the mean squared error of the frequency estimation methods for different n and ε averaged over all datasets. We provide a more detailed overview of the results in Figure 14 in the appendix. From these figures, we observe that most methods have a similar error for given n and ε and that an increase in n or ε typically leads to a decrease in error. However, some frequency oracles show a different behavior: OLH, OUE, Nguyên et al. [Ngu+16], and HM all show a "levelling off" effect where the error

does not decrease further for increasing ε. OUE and OLH even show an increase in error before levelling off for larger ε. Nguyên et al. [Ngu+16] produces roughly the same error for $\varepsilon < 3$ regardless of n. RAPPOR [EPK14] shows another interesting behavior, where the error first decreases with increasing ε reaching a minimum around $\varepsilon = 1$ and then increases again for larger ε. Furthermore, while RAPPOR shows the lowest error among all methods for ε around 1 and $n = 1000$, this advantage diminishes for larger n. For $n = 100\,000$, RAPPOR only performs best for $\varepsilon \approx 1$ and performs worse than the best methods for all other ε.

The other methods show comparable error rates, with a few exceptions: SUE (or Duchi, Wainwright, and Jordan [DWJ13]) and HR perform similarly but do not decrease as fast for large ε as the other methods. l-Subset shows competitive error for large k, but does not scale well with n for smaller ε around 1 and small k (see panels for $n = 10000, 100000$ and $k = 8, 16$ in Figure 16). Direct Encoding / k-RR / O-RR [KBR16][5] behave similarly to the other methods for small d, but show a higher error for larger k. Murakami, Hino, and Sakuma [MHS18] on the other hand benefits from large k and small n and ε and shows substantially smaller error for small ε and n. This effect is stronger for larger k. In terms of their performance, we were unable to see a clear difference between the groups of pure and non-pure frequency oracles.

Comparing the variance of the frequency oracles (see Figure 15 in the appendix) to the MSE (see Figure 14 in the appendix), we see a mismatch between the two. While the variance of most methods stays roughly constant for $\varepsilon > 1$, the MSE of most methods still decreases. Furthermore, for most methods the variance converges to the same method-specific value regardless of n when ε increases. This indicates that the variance is not a good indicator of the error of the frequency estimation methods, although several previous works have only discussed the variance of the frequency estimates as a measure of the methods' utility (see e.g., [Wan+17] and [CMM21]).

5 Practical Considerations

This section presents the general findings of our empirical evaluation and discusses some open topics that the research community must address before local differential privacy can be widely adopted for the estimation of descriptive statistics in practice.

5.1 General Findings

There is an abundance of literature describing methods for estimating various descriptive statistics under LDP (see Section 3). However, practical differences between the methods are often minor, with most methods showing similar performance in our empirical evaluations. For the mean estimation task, the choice of method becomes relevant when $\varepsilon > 1$, where the error rates of the methods start to diverge. Here, the Piecewise method by Wang et al. [Wan+19b] shows the best performance and the Bernoulli-based methods

[5]Note that we were unable to reproduce a reduction in error for O-RR [KBR16] when increasing the number of cohorts, as reported in their paper. For this reason, O-RR is equivalent to k-RR in our evaluation.

[DWJ13; DKY17; Ngu+16; WWR23] show the worst performance, even falling behind the basic Laplace mechanism.

In multi-dimensional mean estimation and variance estimation, our results validate the well-known fact that for multiple queries (or dimensions), splitting the participants by the number of queries is performs better than splitting the privacy budget.

While most methods show similar performance in the frequency estimation task, their differences for different parameters are more complex and do not allow to pick a clear winner for all cases. For small k, k-RR or l-Subset are good choices regardless of n and ε. For larger k (we tested up to $k = 512$), the method by Murakami, Hino, and Sakuma [MHS18] shows the best performance for small ε and n, RAPPOR shows the best performance for $\varepsilon \leq 3$ and l-Subset, k-RR and Murakami, Hino, and Sakuma [MHS18] show the best performance for $\varepsilon > 3$. For guidelines on very large k, refer to Wang et al. [Wan+17] and Cormode, Maddock, and Maple [CMM21].

5.2 Towards Practical Application

When utility is the primary concern, central differential privacy would be the preferred choice for estimating descriptive statistics. However, as discussed in the introduction, LDP offers a better trust model and is more suitable for applications where the data is distributed across multiple parties. In this section, we discuss how this gap may be bridged to make LDP more practical for estimating descriptive statistics. Furthermore, choosing the right privacy budget ε and explaining the privacy guarantee given by LDP to the users of the system are difficult tasks that need to be addressed to ensure the usability of LDP in practice.

5.2.1 Improved Utility and Multiple Queries

Most methods are designed for low-dimensional data or single queries. In practice, datasets may be high-dimensional or contain multiple attributes of different types (e.g., numerical and categorical) thus requiring multiple queries to calculate all statistics of interest. The naive solution for handling multiple queries is to apply the methods for each query separately either with a reduced privacy budget or a reduced number of participants to ensure the overall privacy budget is not exceeded. While this may be acceptable when the number of queries is small, it is not practical when the number of queries is large. In this case, the shuffle model [Bit+17] may provide a path forward.

The shuffle model introduces an additional participant, the shuffler $\mathcal{S}$, into the protocol. This shuffler is a trusted and randomized entity that takes the input values from each party and produces a random permutation of those inputs. The main goal of this shuffling process is to eliminate any trace of information about the original position of each input before the data is made public. Several approaches have been used to achieve shuffling in this model. One such method involves secret sharing, as demonstrated in the work of Balle et al. [Bal+20]. Another strategy uses well-shufflable data structures, such as wedges, as proposed by Imola, Murakami, and Chaudhuri [IMC22]. In addition, shuffling can be accomplished using strings [CZ22] or by employing frequency estimation techniques [LWY22].

5.2.2 Guarantees and Confidence Intervals

We have seen that the errors are often accompanied by large standard deviations, i.e., large differences between individual runs of the randomized algorithms, which can make the interpretation of the resulting statistics difficult as it is unclear how far they deviate from the true value. Here, methods that provide confidence intervals could be beneficial. To the best of our knowledge, only Waudby-Smith, Wu, and Ramdas [WWR23] and Gaboardi, Rogers, and Sheffet [GRS19] discuss methods for estimating confidence intervals for mean estimation and there is no work on confidence intervals for the other statistics discussed in this paper.

5.2.3 Susceptibility to Attacks

While different LDP methods with the same privacy budget ε offer the same worst-case privacy guarantees, they may show a different susceptibility to attacks in practice. Arcolezi et al. [Arc+23] analyze the success rate of re-identification attacks on frequency estimation methods under LDP and find that, given the same privacy budget, different methods show a different success rate. It is therefore important to consider differences in attack susceptibility when choosing an LDP method.

5.2.4 Choosing ε and Improving Usability

Choosing the right privacy budget ε is a difficult task for practitioners as can be seen by the differences in choice for the few known applications [DKY17; EPK14; Dif17]. Fernandes, McIver, and Sadeghi [FMS24] use information theory and quantitative information flow to give an interpretation of ε in the context of LDP. Their work provides an important step towards understanding ε under multiple threat models. However, further work is needed to make these theoretical insights approachable for practitioners. Relatedly, the choice of ε and the guarantee given by LDP is difficult to explain to the end users of the system, which can lead to a lack of trust in the system and a reduction in their willingness to participate. Nanayakkara et al. [Nan+23] is one of the few works to address the issue of explaining ε in the central DP setting. Further work is needed to transfer these ideas to the LDP setting.

6 Conclusion

Local differential privacy (LDP) offers strong privacy guarantees and a trust model for estimating descriptive statistics in distributed settings, which are increasingly relevant in practice. In this SoK, we systematize the literature on LDP for estimating descriptive statistics and provide an extensive empirical comparison of methods for estimating the mean, variance, and frequency of data. Although some open topics remain before LDP can be widely adopted, its use can enhance trust in data analysis and sharing in distributed settings. Our systematization and empirical evaluation serve as a starting point for practitioners to choose the right method for their specific use case and for researchers to focus on the practical aspects of LDP.

Acknoledgement

René Raab acknowledges funding provided by the Federal Ministry for Economic Affairs and Climate Action (BMWK) under Grant No. 68GX21004F (TEAM-X). This work was partially supported by Deutsche Forschungsgemeinschaft as part of the Research and Training Group 2475 "Cybercrime and Forensic Computing" (grant number 393541319/GRK2475/1-2019), grant 442893093, by the state of Bavaria at the Nuremberg Campus of Technology (NCT) which is a research cooperation between the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU) and the Technische Hochschule Nürnberg Georg Simon Ohm (THN), and by the Smart Networks and Services Joint Undertaking (SNS JU) under the European Union's Horizon Europe research and innovation program in the scope of the CONFIDENTIAL6G project under Grant Agreement 101096435, The contents of this publication are the sole responsibility of the authors and do not in any way reflect the views of the EU. The authors gratefully acknowledge the scientific support and HPC resources provided by the Erlangen National High Performance Computing Center (NHR@FAU) of the Friedrich-Alexander-Universität Erlangen-Nürnberg (FAU). The hardware is funded by the German Research Foundation (DFG). We thank the anonymous reviewers and the revision editor for their very helpful comments and suggestions.

References

[AA01] Dakshi Agrawal and Charu C. Aggarwal. "On the Design and Quantification of Privacy Preserving Data Mining Algorithms". In: *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. PODS '01. New York, NY, USA: Association for Computing Machinery, May 2001, pp. 247–255. ISBN: 978-1-58113-361-5. DOI: 10.1145/375551.375602. (Visited on 08/26/2024).

[ACP23] Héber H. Arcolezi, Selene Cerna, and Catuscia Palamidessi. "On the Utility Gain of Iterative Bayesian Update for Locally Differentially Private Mechanisms". In: *Data and Applications Security and Privacy XXXVII*. Ed. by Vijayalakshmi Atluri and Anna Lisa Ferrara. Cham: Springer Nature Switzerland, 2023, pp. 165–183. ISBN: 978-3-031-37586-6. DOI: 10.1007/978-3-031-37586-6_11.

[AFT22] Hilal Asi, Vitaly Feldman, and Kunal Talwar. "Optimal Algorithms for Mean Estimation under Local Differential Privacy". In: *Proceedings of the 39th International Conference on Machine Learning*. PMLR, June 2022, pp. 1046–1056. URL: https://proceedings.mlr.press/v162/asi22b.html (visited on 05/12/2023).

[AH17] Mousumi Akter and Tanzima Hashem. "Computing Aggregates Over Numeric Data with Personalized Local Differential Privacy". In: *Information Security and Privacy*. Ed. by Josef Pieprzyk and Suriadi Suriadi. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017, pp. 249–260. ISBN: 978-3-319-59870-3. DOI: 10.1007/978-3-319-59870-3_14.

[Alv+18a] Mário Alvim et al. "Invited Paper: Local Differential Privacy on Metric Spaces: Optimizing the Trade-Off with Utility". In: *2018 IEEE 31st Computer Security Foundations Symposium (CSF)*. July 2018, pp. 262–267. DOI: 10.1109/CSF.2018.00026. (Visited on 08/26/2024).

[Alv+18b] Mário S. Alvim et al. *Metric-Based Local Differential Privacy for Statistical Applications*. May 2018. DOI: 10.48550/arXiv.1805.01456. arXiv: 1805.01456 [cs]. (Visited on 08/02/2024).

[Amm+23] Tatjana Ammer et al. "A pipeline for the fully automated estimation of continuous reference intervals using real-world data". In: *Scientific Reports* 13.1 (2023), p. 13440. DOI: 10.1038/s41598-023-40561-3. URL: https://doi.org/10.1038/s41598-023-40561-3.

[Arc+23] Héber H. Arcolezi et al. "On the Risks of Collecting Multidimensional Data Under Local Differential Privacy". In: *Proc. VLDB Endow.* 16.5 (Jan. 2023), pp. 1126–1139. ISSN: 2150-8097. DOI: 10.14778/3579075.3579086. (Visited on 07/09/2024).

[ASZ19] Jayadev Acharya, Ziteng Sun, and Huanyu Zhang. "Hadamard Response: Estimating Distributions Privately, Efficiently, and with Little Communication". In: *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. PMLR, Apr. 2019, pp. 1120–1129. URL: https://proceedings.mlr.press/v89/acharya19a.html (visited on 01/19/2024).

[Bal+20] Borja Balle et al. "Private Summation in the Multi-Message Shuffle Model". In: *ACM CCS 2020: 27th Conference on Computer and Communications Security*. Ed. by Jay Ligatti et al. Virtual Event, USA: ACM Press, 2020, pp. 657–676. DOI: 10.1145/3372297.3417242.

[Bas+20] Raef Bassily et al. "Practical Locally Private Heavy Hitters". In: *Journal of Machine Learning Research* 21.16 (2020), pp. 1–42. ISSN: 1533-7928. URL: http://jmlr.org/papers/v21/18-786.html (visited on 03/18/2024).

[BGS24] Pascal Berrang, Paul Gerhart, and Dominique Schröder. "Measuring Conditional Anonymity — A Global Study". In: *Proceedings on Privacy Enhancing Technologies* 2024.4 (2024), pp. 947–966. ISSN: 2299-0984. DOI: 10.56553/popets-2024-0150.

[Bho+19] Abhishek Bhowmick et al. *Protection Against Reconstruction and Its Applications in Private Federated Learning*. June 2019. DOI: 10.48550/arXiv.1812.00984. arXiv: 1812.00984 [cs, stat]. (Visited on 09/13/2023).

[Bit+17] Andrea Bittau et al. "Prochlo: Strong Privacy for Analytics in the Crowd". In: *Proceedings of the 26th Symposium on Operating Systems Principles*. Oct. 2017, pp. 441–459. DOI: 10.1145/3132747.3132769. arXiv: 1710.00901 [cs]. (Visited on 05/11/2023).

[BK96] Barry Becker and Ronny Kohavi. *Adult*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5XW20. 1996.

[Bla15] Martin Bland. *An Introduction to Medical Statistics*. Fourth edition. Oxford Medical Publications. Oxford: Oxford University Press, 2015. ISBN: 978-0-19-958992-0.

[BP24] Sayan Biswas and Catuscia Palamidessi. "PRIVIC: A Privacy-Preserving Method for Incremental Collection of Location Data". In: *Proceedings on Privacy Enhancing Technologies* (2024). ISSN: 2299-0984. DOI: 10.56553/popets-2024-0033. (Visited on 08/02/2024).

[BS15a] Raef Bassily and Adam Smith. "Local, Private, Efficient Protocols for Succinct Histograms". In: *Proceedings of the Forty-Seventh Annual ACM Symposium on Theory of Computing*. STOC '15. New York, NY, USA: Association for Computing Machinery, June 2015, pp. 127–135. ISBN: 978-1-4503-3536-2. DOI: 10.1145/2746539.2746632. (Visited on 09/08/2023).

[BS15b] Raef Bassily and Adam D. Smith. "Local, Private, Efficient Protocols for Succinct Histograms". In: *47th Annual ACM Symposium on Theory of Computing*. Ed. by Rocco A. Servedio and Ronitt Rubinfeld. Portland, OR, USA: ACM Press, 2015, pp. 127–135. DOI: 10.1145/2746539.2746632.

[Cit02] City of New York - Taxi and Limousine Commission. *TLC Trip Record Data*. c2024. URL: https://www.nyc.gov/site/tlc/about/tlc-trip-record-data.page (visited on 04/16/2024).

[CKS18] Graham Cormode, Tejas Kulkarni, and Divesh Srivastava. "Marginal Release Under Local Differential Privacy". In: *Proceedings of the 2018 International Conference on Management of Data*. Houston TX USA: ACM, May 2018, pp. 131–146. ISBN: 978-1-4503-4703-7. DOI: 10.1145/3183713.3196906. (Visited on 05/31/2023).

[CKS19] Graham Cormode, Tejas Kulkarni, and Divesh Srivastava. "Answering Range Queries under Local Differential Privacy". In: *Proceedings of the VLDB Endowment* 12.10 (June 2019), pp. 1126–1138. ISSN: 2150-8097. DOI: 10.14778/3339490.3339496. (Visited on 07/07/2023).

[CMM21] Graham Cormode, Samuel Maddock, and Carsten Maple. "Frequency Estimation under Local Differential Privacy". In: *Proceedings of the VLDB Endowment* 14.11 (July 2021), pp. 2046–2058. ISSN: 2150-8097. DOI: 10.14778/3476249.3476261. (Visited on 07/07/2023).

[CV17] Bennett Cyphers and Kalyan Veeramachaneni. "AnonML: Locally Private Machine Learning over a Network of Peers". In: *2017 IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. Oct. 2017, pp. 549–560. DOI: 10.1109/DSAA.2017.80. (Visited on 03/13/2024).

[CZ22] Albert Cheu and Maxim Zhilyaev. "Differentially Private Histograms in the Shuffle Model from Fake Users". In: *2022 IEEE Symposium on Security and Privacy*. San Francisco, CA, USA: IEEE Computer Society Press, 2022, pp. 440–457. DOI: 10.1109/SP46214.2022.9833614.

[Dia+20] Xinrong Diao et al. "PrivGMM: Probability Density Estimation with Local Differential Privacy". In: *Database Systems for Advanced Applications: 25th International Conference, DASFAA 2020, Jeju, South Korea, September 24–27, 2020, Proceedings, Part I*. Berlin, Heidelberg: Springer-Verlag, Sept. 2020, pp. 105–121. ISBN: 978-3-030-59409-1. DOI: 10.1007/978-3-030-59410-7_7. (Visited on 07/07/2023).

[Dif17] Differential Privacy Team, Apple. *Learning with Privacy at Scale*. Tech. rep. 2017. URL: https://docs-assets.developer.apple.com/ml-research/papers/learning-with-privacy-at-scale.pdf.

[Din+18] Bolin Ding et al. "Comparing Population Means Under Local Differential Privacy: With Significance and Power". In: *Proceedings of the AAAI Conference on Artificial Intelligence* 32.1 (Apr. 2018). ISSN: 2374-3468. DOI: 10.1609/aaai.v32i1.11301. (Visited on 06/15/2023).

[DJW14] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. *Local Privacy, Data Processing Inequalities, and Statistical Minimax Rates*. Aug. 2014. arXiv: 1302.3203 [cs, math, stat]. URL: http://arxiv.org/abs/1302.3203 (visited on 06/09/2023).

[DJW18] John C. Duchi, Michael I. Jordan, and Martin J. Wainwright. "Minimax Optimal Procedures for Locally Private Estimation". In: *Journal of the American Statistical Association* 113.521 (Jan. 2018), pp. 182–201. ISSN: 0162-1459. DOI: 10.1080/01621459.2017.1389735. (Visited on 05/12/2023).

[DKY17] Bolin Ding, Janardhan Kulkarni, and Sergey Yekhanin. "Collecting Telemetry Data Privately". In: *Advances in Neural Information Processing Systems*. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper_files/paper/2017/hash/253614bbac999b38b5b60cae531c4969-Abstract.html (visited on 07/04/2023).

[DR14] Cynthia Dwork and Aaron Roth. "The Algorithmic Foundations of Differential Privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (Aug. 2014), pp. 211–407. ISSN: 1551-305X. DOI: 10/gcgmcw. (Visited on 10/29/2021).

[Du+21] Linkang Du et al. "AHEAD: Adaptive Hierarchical Decomposition for Range Query under Local Differential Privacy". In: *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*. CCS '21. New York, NY, USA: Association for Computing Machinery, Nov. 2021, pp. 1266–1288. ISBN: 978-1-4503-8454-4. DOI: 10.1145/3460120.3485668. (Visited on 08/18/2023).

[DWJ13] John Duchi, Martin J Wainwright, and Michael I Jordan. "Local Privacy and Minimax Bounds: Sharp Rates for Probability Estimation". In: *Advances in Neural Information Processing Systems*. Vol. 26. Curran Associates, Inc., 2013. URL: https://papers.nips.cc/paper_files/paper/2013/hash/5807a685d1a9ab3b599035bc566ce2b9-Abstract.html (visited on 09/04/2023).

[Dwo+06] Cynthia Dwork et al. “Calibrating Noise to Sensitivity in Private Data Analysis”. In: *Theory of Cryptography*. Ed. by Shai Halevi and Tal Rabin. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer, 2006, pp. 265–284. ISBN: 978-3-540-32732-5. DOI: 10.1007/11681878_14.

[EP20] Ehab ElSalamouny and Catuscia Palamidessi. “Generalized Iterative Bayesian Update and Applications to Mechanisms for Privacy Protection”. In: *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. Sept. 2020, pp. 490–507. DOI: 10.1109/EuroSP48549.2020.00038. (Visited on 08/01/2024).

[EPK14] Úlfar Erlingsson, Vasyl Pihur, and Aleksandra Korolova. “RAPPOR: Randomized Aggregatable Privacy-Preserving Ordinal Response”. In: *ACM CCS 2014: 21st Conference on Computer and Communications Security*. Ed. by Gail-Joon Ahn, Moti Yung, and Ninghui Li. Scottsdale, AZ, USA: ACM Press, 2014, pp. 1054–1067. DOI: 10.1145/2660267.2660348.

[FDM19] Natasha Fernandes, Mark Dras, and Annabelle McIver. “Generalised Differential Privacy for Text Document Processing”. In: *Principles of Security and Trust*. Ed. by Flemming Nielson and David Sands. Cham: Springer International Publishing, 2019, pp. 123–148. ISBN: 978-3-030-17138-4. DOI: 10.1007/978-3-030-17138-4_6.

[FMS24] Natasha Fernandes, Annabelle McIver, and Parastoo Sadeghi. “Explaining Epsilon in Local Differential Privacy through the Lens of Quantitative Information Flow”. In: *2024 IEEE 37th Computer Security Foundations Symposium (CSF)*. IEEE Computer Society, Apr. 2024, pp. 175–188. ISBN: 9798350362039. DOI: 10.1109/CSF61375.2024.00012. (Visited on 07/19/2024).

[FPE16] Giulia Fanti, Vasyl Pihur, and Úlfar Erlingsson. “Building a RAPPOR with the Unknown: Privacy-Preserving Learning of Associations and Data Dictionaries”. In: *Proceedings on Privacy Enhancing Technologies* 2016.3 (July 2016), pp. 41–61. ISSN: 2299-0984. DOI: 10.1515/popets-2016-0015. (Visited on 03/11/2024).

[FYS22] Kazuto Fukuchi, Chia-Mu Yu, and Jun Sakuma. “Locally Differentially Private Minimum Finding”. In: *IEICE Transactions on Information and Systems* E105.D.8 (2022), pp. 1418–1430. DOI: 10.1587/transinf.2021EDP7187.

[Gal+23] Filippo Galli et al. “Advancing Personalized Federated Learning: Group Privacy, Fairness, and Beyond”. In: *SN Computer Science* 4.6 (Oct. 2023), p. 831. ISSN: 2661-8907. DOI: 10.1007/s42979-023-02292-0. (Visited on 08/26/2024).

[GRS19] Marco Gaboardi, Ryan Rogers, and Or Sheffet. “Locally Private Mean Estimation: Z-Test and Tight Confidence Intervals”. In: *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*. PMLR, Apr. 2019, pp. 2545–2554. URL: https://proceedings.mlr.press/v89/gaboardi19a.html (visited on 07/07/2023).

[Gu+20] Xiaolan Gu et al. "PCKV: Locally Differentially Private Correlated Key-Value Data Collection with Optimized Utility". In: *USENIX Security 2020: 29th USENIX Security Symposium*. Ed. by Srdjan Capkun and Franziska Roesner. USENIX Association, 2020, pp. 967–984.

[IMC22] Jacob Imola, Takao Murakami, and Kamalika Chaudhuri. "Differentially Private Triangle and 4-Cycle Counting in the Shuffle Model". In: *ACM CCS 2022: 29th Conference on Computer and Communications Security*. Ed. by Heng Yin et al. Los Angeles, CA, USA: ACM Press, 2022, pp. 1505–1519. DOI: 10.1145/3548606.3560659.

[Jos+19a] Matthew Joseph et al. "Locally Private Gaussian Estimation". In: *Advances in Neural Information Processing Systems*. Vol. 32. Curran Associates, Inc., 2019. URL: https://papers.nips.cc/paper_files/paper/2019/hash/a588a6199feff5ba48402883d9b72700-Abstract.html (visited on 08/22/2023).

[Jos+19b] Matthew Joseph et al. "The Role of Interactivity in Local Differential Privacy". In: *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*. Nov. 2019, pp. 94–105. DOI: 10.1109/FOCS.2019.00015.

[Kai+20] Georgios A. Kaissis et al. "Secure, privacy-preserving and federated machine learning in medical imaging". In: *Nature Machine Intelligence* 2.6 (2020), pp. 305–311. DOI: 10.1038/s42256-020-0186-1. URL: https://doi.org/10.1038/s42256-020-0186-1.

[Kas+11] Shiva Prasad Kasiviswanathan et al. "What Can We Learn Privately?" In: *SIAM Journal on Computing* 40.3 (Jan. 2011), pp. 793–826. ISSN: 0097-5397. DOI: 10.1137/090756090. (Visited on 05/05/2023).

[KBR16] Peter Kairouz, Keith Bonawitz, and Daniel Ramage. "Discrete Distribution Estimation under Local Privacy". In: *Proceedings of The 33rd International Conference on Machine Learning*. PMLR, June 2016, pp. 2436–2444. URL: https://proceedings.mlr.press/v48/kairouz16.html (visited on 02/09/2024).

[KOV14] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. "Extremal Mechanisms for Local Differential Privacy". In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc., 2014. URL: https://papers.nips.cc/paper_files/paper/2014/hash/86df7dcfd896fcaf2674f757a2463eba-Abstract.html (visited on 03/18/2024).

[KOV16] Peter Kairouz, Sewoong Oh, and Pramod Viswanath. "Extremal Mechanisms for Local Differential Privacy". In: *Journal of Machine Learning Research* 17.17 (2016), pp. 1–51. ISSN: 1533-7928. URL: http://jmlr.org/papers/v17/15-135.html (visited on 07/12/2023).

[Li+20] Zitao Li et al. "Estimating Numerical Distributions under Local Differential Privacy". In: *Proceedings of the 2020 ACM SIGMOD International Conference on Management of Data*. Portland OR USA: ACM, June 2020, pp. 621–635. ISBN: 978-1-4503-6735-6. DOI: 10.1145/3318464.3389700. (Visited on 05/25/2023).

[LWY22] Qiyao Luo, Yilei Wang, and Ke Yi. "Frequency Estimation in the Shuffle Model with Almost a Single Message". In: *ACM CCS 2022: 29th Conference on Computer and Communications Security*. Ed. by Heng Yin et al. Los Angeles, CA, USA: ACM Press, 2022, pp. 2219–2232. DOI: 10.1145/3548606.3560608.

[MHS18] Takao Murakami, Hideitsu Hino, and Jun Sakuma. "Toward Distribution Estimation under Local Differential Privacy with Small Samples". In: *Proceedings on Privacy Enhancing Technologies* 2018.3 (July 2018), pp. 84–104. DOI: 10.1515/popets-2018-0022.

[MTH90] Chris Meek, Bo Thiesson, and David Heckerman. *US Census Data (1990)*. UCI Machine Learning Repository. DOI: https://doi.org/10.24432/C5VP42. 1990.

[Nan+23] Priyanka Nanayakkara et al. "What Are the Chances? Explaining the Epsilon Parameter in Differential Privacy". In: *32nd USENIX Security Symposium (USENIX Security 23)*. 2023, pp. 1613–1630. ISBN: 978-1-939133-37-3. URL: https://www.usenix.org/conference/usenixsecurity23/presentation/nanayakkara (visited on 04/26/2024).

[Ngu+16] Thông T. Nguyên et al. *Collecting and Analyzing Data from Smart Device Users with Local Differential Privacy*. June 2016. DOI: 10.48550/arXiv.1606.05053. arXiv: 1606.05053 [cs]. (Visited on 07/06/2023).

[PP20] Janet L. Peacock and Phil J. Peacock. *Oxford Handbook of Medical Statistics*. 2nd ed. Oxford University Press, May 2020. ISBN: 978-0-19-874358-3 978-0-19-180320-8. DOI: 10.1093/med/9780198743583.001.0001. (Visited on 01/10/2024).

[Qin+16] Zhan Qin et al. "Heavy Hitter Estimation over Set-Valued Data with Local Differential Privacy". In: *ACM CCS 2016: 23rd Conference on Computer and Communications Security*. Ed. by Edgar R. Weippl et al. Vienna, Austria: ACM Press, 2016, pp. 192–203. DOI: 10.1145/2976749.2978409.

[Raa+23] René Raab et al. "Federated Electronic Health Records for the European Health Data Space". In: *The Lancet Digital Health* 5.11 (Nov. 2023), e840–e847. ISSN: 2589-7500. DOI: 10.1016/S2589-7500(23)00156-5. (Visited on 11/07/2023).

[Ren+18] Xuebin Ren et al. "LoPub: High-Dimensional Crowdsourced Data Publication With Local Differential Privacy". In: *IEEE Transactions on Information Forensics and Security* 13.9 (Sept. 2018), pp. 2151–2166. ISSN: 1556-6021. DOI: 10.1109/TIFS.2018.2812146.

[Swe97] Latanya Sweeney. "Weaving Technology and Policy Together to Maintain Confidentiality". In: *The Journal of Law, Medicine & Ethics* 25.2-3 (June 1997), pp. 98–110. ISSN: 1073-1105. DOI: 10.1111/j.1748-720X.1997.tb01885.x. (Visited on 05/29/2024).

[SXY21] Zixuan Shen, Zhihua Xia, and Peipeng Yu. "PLDP: Personalized Local Differential Privacy for Multidimensional Data Aggregation". In: *Security and Communication Networks* 2021.1 (2021), p. 6684179. ISSN: 1939-0122. DOI: 10.1155/2021/6684179. (Visited on 08/02/2024).

[Wan+16] Shaowei Wang et al. *Mutual Information Optimally Local Private Discrete Distribution Estimation*. July 2016. DOI: 10.48550/arXiv.1607.08025. arXiv: 1607.08025 [cs, math]. (Visited on 01/19/2024).

[Wan+17] Tianhao Wang et al. "Locally Differentially Private Protocols for Frequency Estimation". In: *USENIX Security 2017: 26th USENIX Security Symposium*. Ed. by Engin Kirda and Thomas Ristenpart. Vancouver, BC, Canada: USENIX Association, 2017, pp. 729–745.

[Wan+19a] Ning Wang et al. "Collecting and Analyzing Multidimensional Data with Local Differential Privacy". In: *2019 IEEE 35th International Conference on Data Engineering (ICDE)*. IEEE Computer Society, Apr. 2019, pp. 638–649. ISBN: 978-1-5386-7474-1. DOI: 10.1109/ICDE.2019.00063. (Visited on 08/22/2023).

[Wan+19b] Tianhao Wang et al. "Answering Multi-Dimensional Analytical Queries under Local Differential Privacy". In: *Proceedings of the 2019 International Conference on Management of Data*. Amsterdam Netherlands: ACM, June 2019, pp. 159–176. ISBN: 978-1-4503-5643-5. DOI: 10.1145/3299869.3319891. (Visited on 05/25/2023).

[Wan+20] Teng Wang et al. "A Comprehensive Survey on Local Differential Privacy toward Data Statistics and Analysis". In: *Sensors* 20.24 (Jan. 2020), p. 7030. ISSN: 1424-8220. DOI: 10.3390/s20247030.

[War65] Stanley L. Warner. "Randomized Response: A Survey Technique for Eliminating Evasive Answer Bias". In: *Journal of the American Statistical Association* 60.309 (Mar. 1965), pp. 63–69. ISSN: 0162-1459. DOI: 10.1080/01621459.1965.10480775. (Visited on 05/05/2023).

[WC13] Weiran Wang and Miguel Á Carreira-Perpiñán. *Projection onto the Probability Simplex: An Efficient Algorithm with a Simple Proof, and an Application*. Sept. 2013. DOI: 10.48550/arXiv.1309.1541. arXiv: 1309.1541 [cs, math, stat]. (Visited on 05/15/2024).

[WLJ21] Tianhao Wang, Ninghui Li, and Somesh Jha. "Locally Differentially Private Heavy Hitter Identification". In: *IEEE Transactions on Dependable and Secure Computing* 18.2 (Mar. 2021), pp. 982–993. ISSN: 1941-0018. DOI: 10.1109/TDSC.2019.2927695. (Visited on 03/18/2024).

[WWR23] Ian Waudby-Smith, Steven Wu, and Aaditya Ramdas. "Nonparametric Extensions of Randomized Response for Private Confidence Sets". In: *Proceedings of the 40th International Conference on Machine Learning*. PMLR, July 2023, pp. 36748–36789. URL: https://proceedings.mlr.press/v202/waudby-smith23a.html (visited on 08/22/2023).

[Xio+20] Xingxing Xiong et al. "A Comprehensive Survey on Local Differential Privacy". In: *Security and Communication Networks* 2020 (Oct. 2020), e8829523. ISSN: 1939-0114. DOI: 10.1155/2020/8829523.

[XZW21a] Qiao Xue, Youwen Zhu, and Jian Wang. "Joint Distribution Estimation and Naïve Bayes Classification Under Local Differential Privacy". In: *IEEE Transactions on Emerging Topics in Computing* 9.4 (Oct. 2021), pp. 2053–2063. ISSN: 2168-6750. DOI: 10.1109/TETC.2019.2959581. (Visited on 03/13/2024).

[XZW21b] Qiao Xue, Youwen Zhu, and Jian Wang. "Mean Estimation over Numeric Data with Personalized Local Differential Privacy". In: *Frontiers of Computer Science* 16.3 (Sept. 2021), p. 163806. ISSN: 2095-2236. DOI: 10.1007/s11704-020-0103-0. (Visited on 07/07/2023).

[Yan+20] Jianyu Yang et al. "Answering Multi-Dimensional Range Queries under Local Differential Privacy". In: *Proceedings of the VLDB Endowment* 14.3 (Nov. 2020), pp. 378–390. ISSN: 2150-8097. DOI: 10.14778/3430915.3430927. (Visited on 07/07/2023).

[Yan+24] Mengmeng Yang et al. "Local Differential Privacy and Its Applications: A Comprehensive Survey". In: *Computer Standards & Interfaces* 89 (Apr. 2024), p. 103827. ISSN: 0920-5489. DOI: 10.1016/j.csi.2023.103827. (Visited on 02/27/2024).

[YB17] Min Ye and Alexander Barg. "Optimal Schemes for Discrete Distribution Estimation under Local Differential Privacy". In: *2017 IEEE International Symposium on Information Theory (ISIT)*. Aachen, Germany: IEEE, June 2017, pp. 759–763. ISBN: 978-1-5090-4096-4. DOI: 10.1109/ISIT.2017.8006630. (Visited on 01/19/2024).

[Ye+19] Qingqing Ye et al. "PrivKV: Key-Value Data Collection with Local Differential Privacy". In: *2019 IEEE Symposium on Security and Privacy*. San Francisco, CA, USA: IEEE Computer Society Press, 2019, pp. 317–331. DOI: 10.1109/SP.2019.00018.

[Zha+18] Zhikun Zhang et al. "CALM: Consistent Adaptive Local Marginal for Marginal Release under Local Differential Privacy". In: *ACM CCS 2018: 25th Conference on Computer and Communications Security*. Ed. by David Lie et al. Toronto, ON, Canada: ACM Press, 2018, pp. 212–229. DOI: 10.1145/3243734.3243742.

[Zho+22] Mingxun Zhou et al. "Locally Differentially Private Sparse Vector Aggregation". In: *2022 IEEE Symposium on Security and Privacy*. San Francisco, CA, USA: IEEE Computer Society Press, 2022, pp. 422–439. DOI: 10.1109/SP46214.2022.9833635.

[Zie+20] Jakob Zierk et al. "Reference Interval Estimation from Mixed Distributions using Truncation Points and the Kolmogorov-Smirnov Distance (kosmic)". In: *Scientific Reports* 10.1 (2020), p. 1704. DOI: 10.1038/s41598-020-58749-2. URL: https://doi.org/10.1038/s41598-020-58749-2.

A Notation and Methods Overview

Table 4 provides an overview of the notation used in this paper. Tables 5 and 6 provide an overview of the methods for estimating descriptive statistics under local differential privacy. Table 9 gives more details on the methods for estimating the mean of Gaussian data.

Table 4: Notation used in this paper. Further notation may be used in parts of the paper and is defined there.

Symbol	**Description**
	General Notation
n	Number of participants/clients
$[n]$	The set $\{1, 2, \ldots, n\}$
ε	Privacy budget of ε-(L)DP
δ	Privacy parameter ("privacy failure")
x_i	Private input of participant i
z_i	Noisy output of participant i
d	Dimension of the data
$\mathcal{A}$	Randomized algorithm
$B[i]$	Entry at index i in vector B
	Mean Estimation
$\bar{x}$	True sample mean
μ	True population mean
$\hat{\mu}$	Estimated (sample or population) mean
$\hat{\sigma}^2$	Estimated variance
	Frequency Estimation
$\mathcal{D}$	Domain of the data
k	Domain size
$l \leq k$	Subset size (for the l-Subset mechanism)
$\hat{f}(x)$	Estimated frequency of x
$f(x)$	True frequency of x

Table 5: Overview of the methods for estimating descriptive statistics under local differential privacy. Continued in Tables 6, 7, and 8.

Mean Estimation	
Method	**Summary**
Laplace Mechanism	Add noise from Laplace distribution to the mean
[DWJ13; DJW18] – ℓ_2	Bernoulli-based mechanism for data in an ℓ_2 ball
[DWJ13] – ℓ_∞	Bernoulli-based mechanism for data in an ℓ_∞ ball
[DWJ13] – 1-sparse	Bernoulli-based mechanism for 1-sparse data (only one non-zero entry)
[Ngu+16]	Bernoulli-based mechanism for mean estimation
[DKY17]	Bernoulli-based mechanism for mean estimation
[Wan+19b]	Piecewise and Hybrid mechanisms for mean estimation
[WWR23]	Bernoulli-based mechanism for mean estimation

Table 6: Overview of the methods for estimating descriptive statistics under local differential privacy. [Continuation of Table 5].

Frequency Estimation	
Method	**Summary**
Randomized Response [War65; Wan+17]	Randomized response mechanism for frequency estimation
Direct Encoding / k-ary randomized response [KBR16]	Randomized response for non-binary data
Histogram Encoding [Wan+17]	Encode data as one-hot vectors and add Laplace noise
Unary Encoding [DWJ13]	Encode data as one-hot vectors and independently flip bits
Symmetric Unary Encoding / Basic RAPPOR [EPK14]	like Unary Encoding, but with a specific probability
Optimized Unary Encoding [Wan+17]	like Unary Encoding, but with optimized probability
RAPPOR [EPK14]	Apply randomized response to a Bloom filter
Local Hashing [Wan+17]	Apply a hash function to the data before using direct encoding
Fast Local Hashing [CMM21]	Local Hashing with heuristics
Hadamard Mechanism [Bas+20]	Respond with random Hadamard coefficient
Hadamard Response [ASZ19]	Randomly choose a positive or negative Hadamard coefficient to report
Optimized Randomized Response [KBR16]	Randomized response with cohorts and hash functions
l-Subset [YB17; Wan+16]	Submit a subset of the domain of size $l < k$ to the aggregator
Nguyên et al. [Ngu+16]	Encoding using an orthogonal matrix
ElSalamouny and Palamidessi [EP20]	Postprocessing using Iterative Bayesian Update (IBU) to enable best accuracy. Optimal for any obfuscation mechanism.
Murakami, Hino, and Sakuma [MHS18]	Postprocessing using IBU – designed to cope with small samples
Histogram Estimation	
Method	**Summary**
Duchi, Wainwright, and Jordan [DWJ13]	Encode bins as one-hot vector and apply the Laplace mechanism
Ding, Kulkarni, and Yekhanin [DKY17]	Randomly sample buckets and respond with randomly flipped bits to indicate whether the bucket was the correct one

Table 7: Overview of the methods for estimating descriptive statistics under local differential privacy. [Continuation of Tables 5 and 6].

Distribution Estimation	
Method	**Summary**
Duchi, Jordan, and Wainwright [DJW18]	Estimator based on orthogonal series expansion
Diao et al. [Dia+20]	Model data distribution as a Gaussian mixture model – only approximately LDP
Li et al. [Li+20]	Square wave mechanism – conceptually similar to the Piecewise mechanism for mean estimation [Wan+19a]

Table 8: Overview of the methods for estimating descriptive statistics under local differential privacy [Continuation of Tables 5, 6, and 7].

Contingency Tables & Marginal Tables	
Method	**Summary**
Fanti, Pihur, and Erlingsson [FPE16]	Full contingency table for 2 categorical variables using Expectation Maximization
Ren et al. [Ren+18]	Fixed k-way marginal for d categorical variables using Expectation Maximization / Lasso regression
Cormode, Kulkarni, and Srivastava [CKS18]	All k-way marginals for d binary variables using a Hadamard transform on private data
Zhang et al. [Zha+18]	All k-way marginals for d categorical using Entropy Maximization + Frequency Oracle
Xue, Zhu, and Wang [XZW21a]	Joint distribution for 2 categorical variables based on the k-subset mechanism [Wan+16; YB17]
Range Queries	
Method	**Summary**
Cormode, Kulkarni, and Srivastava [CKS19]	Range queries based on hierarchical histograms
Wang et al. [Wan+19b]	Range queries based on subintervals stored as nodes in a tree structure
Li et al. [Li+20]	Range queries based on the distribution estimation method in the same paper
Yang et al. [Yan+20]	Range queries based on multi-dimensional (sub-)grids
Du et al. [Du+21]	Range queries based on an adaptive hierarchical grid structure
Order Statistics	
Method	**Summary**
Cyphers and Veeramachaneni [CV17]	Find the median using binary search – no analysis of error
Gaboardi, Rogers, and Sheffet [GRS19]	Estimate quantiles based on binary search
Fukuchi, Yu, and Sakuma [FYS22]	Find the minimum/maximum based on binary search
Duchi, Jordan, and Wainwright [DJW18]	Find the median based on stochastic gradient descent
Cormode, Kulkarni, and Srivastava [CKS19]	Find any quantile using range queries

Table 9: Comparison of mean estimation mechanisms for Gaussian data.

Algorithm	**Input Range**	**Error**	**Rounds**	**LDP Type**
Gaboardi, Rogers, and Sheffet [GRS19]	$x_i \sim \mathcal{N}(\mu, \sigma^2), \mu \in [-R, R]$	with prob. $1-\beta$, $\mu \in I$, $\lvert I\rvert = \ldots$		(ε, δ)
- KnownVar	- known σ	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{1}{\beta}\right)\log\left(\frac{n}{\beta}\right)\log\left(\frac{1}{\delta}\right)}\right)$	2	
- UnkVar	- $\sigma \in [\sigma_{\min}, \sigma_{\max}], \sigma_{\max} \leq 2R$	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{1}{\beta}\right)\log\left(\frac{n}{\beta}\right)\log\left(\frac{1}{\delta}\right)}\right)$	$\Omega(\log(\frac{R}{\sigma_{\min}}))$	
Joseph et al. [Jos+19a]	$x_i \sim \mathcal{N}(\mu, \sigma^2)$, $\mu = O\left(2^{n\varepsilon^2/\log(n/\beta)}\right)$	with prob. $1-\beta$, $\lvert\hat{\mu}-\mu\rvert = \ldots$		ε
- KVGausstimate	- known σ	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{1}{\beta}\right)}\right)$	2	
- 1RoundKVGausstimate	- known σ	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{1}{\beta}\right)\sqrt{\log(n)}}\right)$	1	
- UVGausstimate	- $\sigma \in [\sigma_{\min}, \sigma_{\max}]$	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{1}{\beta}\right)\log(n)}\right)$	2	
- 1RoundUVGausstimate	- $\sigma \in [\sigma_{\min}, \sigma_{\max}]$	$O\left(\frac{\sigma}{\varepsilon}\sqrt{\frac{1}{n}\log\left(\frac{\sigma_{\max}}{\sigma_{\min}}+1\right)\log\left(\frac{1}{\beta}\right)\log^{3/2}(n)}\right)$	1	

B Proofs

Proof of Proposition 1. We claim that all Bernoulli-based mechanisms M_{Du}, M_N, M_D, and M_W are equivalent for the mean estimation of 1-dimensional input. The fact, that M_W and M_D are equivalent (for default parameters of M_W), has already been shown by Waudby-Smith, Wu, and Ramdas [WWR23]. We therefore only need to show that M_{Du} and M_N are equivalent and that M_N and M_D are equivalent.

We first show that M_N and M_D are equivalent. We begin, by recalling the definitions of the mechanisms M_D and M_N and their mean estimators.

The mechanism M_D by Ding, Kulkarni, and Yekhanin [DKY17] takes inputs $x_i \in [0, m]$ and outputs $z_i \in \{0, 1\}$. The z_i are sampled from a Bernoulli distribution as

$$z_i \sim \text{Bern}\left(p\right), p = \frac{1}{e^\varepsilon + 1} + \frac{x_i}{m}\frac{e^\varepsilon - 1}{e^\varepsilon + 1}.$$

The mean estimator $\hat{\mu}$ is defined as

$$\hat{\mu} = \frac{m}{n}\sum_{i=1}^{n} \frac{z_i \cdot (e^\varepsilon + 1) - 1}{e^\varepsilon - 1}.$$

The mechanism M_N by Nguyên et al. [Ngu+16] (in the 1-dimensional case) takes inputs $x_i' \in [-1, 1]$ and outputs $z_i' \in \{0, 1\}$. The z_i are sampled from a Bernoulli distribution as

$$z_i' \sim \text{Bern}\left(p'\right), p' = \frac{x_i'(e^\varepsilon - 1) + e^\varepsilon + 1}{2e^\varepsilon + 2}.$$

The participants then respond with

$$u_i' = \begin{cases} \frac{e^\varepsilon+1}{e^\varepsilon-1} & \text{if } z_i' = 1 \\ -\frac{e^\varepsilon+1}{e^\varepsilon-1} & \text{if } z_i' = 0 \end{cases}.$$

The mean estimator $\hat{\mu}'$ is defined as

$$\hat{\mu}' = \frac{1}{n}\sum_{i=1}^{n} u_i'.$$

The authors mention that the participants can also directly respond with z_i' instead of u_i' as this can be calculated by the aggregator. We can rewrite u_i' as $u_i' = (2z_i' - 1)\frac{e^\varepsilon+1}{e^\varepsilon-1}$ and get for the mean estimator

$$\hat{\mu}' = \frac{1}{n}\sum_{i=1}^{n}(2z_i' - 1)\frac{e^\varepsilon + 1}{e^\varepsilon - 1}.$$

We now show that both mechanisms sample from the same distribution if the input is transformed accordingly. We define the transformation $T : [0, m] \to [-1, 1]$ as

$$T(x) = 2\frac{x}{m} - 1.$$

We can then rewrite the probability p' as

$$\begin{aligned}
p' &= \frac{T(x_i)(e^\varepsilon - 1) + e^\varepsilon + 1}{2e^\varepsilon + 2} \\
&= \frac{\left(2\frac{x_i}{m} - 1\right)(e^\varepsilon - 1) + e^\varepsilon + 1}{2e^\varepsilon + 2} \\
&= \frac{2\frac{x_i}{m}(e^\varepsilon - 1) - e^\varepsilon + 1 + e^\varepsilon + 1}{2e^\varepsilon + 2} \\
&= \frac{2\frac{x_i}{m}(e^\varepsilon - 1) + 2}{2e^\varepsilon + 2} \\
&= \frac{2\left(\frac{x_i}{m}(e^\varepsilon - 1) + 1\right)}{2(e^\varepsilon + 1)} \\
&= \frac{1}{e^\varepsilon + 1} + \frac{x_i}{m}\frac{e^\varepsilon - 1}{e^\varepsilon + 1} \\
&= p.
\end{aligned}$$

Therefore, the mechanisms M_D and M_N sample from the same distribution if the input is transformed accordingly.

We now show that the mean estimators are equivalent if we transform their outputs accordingly. We take the mean estimator $\hat{\mu}$ and transform the output from [0, m] to [-1,1] using T.

$$\begin{aligned}
T(\hat{\mu}) = 2\frac{\hat{\mu}}{m} - 1 &= 2\frac{\frac{m}{n}\sum_{i=1}^{n}\frac{z_i\cdot(e^\varepsilon+1)-1}{e^\varepsilon-1}}{m} - 1 \\
&= \frac{2}{n}\left(\sum_{i=1}^{n}\frac{z_i\cdot(e^\varepsilon+1)-1}{e^\varepsilon-1}\right) - 1 \\
&= \frac{1}{n}\left(\sum_{i=1}^{n}2\frac{z_i\cdot(e^\varepsilon+1)-1}{e^\varepsilon-1}\right) - \frac{1}{n}\sum_{i=1}^{n}1 \\
&= \frac{1}{n}\sum_{i=1}^{n}2\frac{z_i\cdot(e^\varepsilon+1)-1}{e^\varepsilon-1} - 1 \\
&= \frac{1}{n}\sum_{i=1}^{n}\left(\frac{2z_i\cdot(e^\varepsilon+1)-2}{e^\varepsilon-1} - 1\right) \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{2z_i\cdot(e^\varepsilon+1)-2-(e^\varepsilon-1)}{e^\varepsilon-1} \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{2z_i\cdot(e^\varepsilon+1)-e^\varepsilon-1}{e^\varepsilon-1} \\
&= \frac{1}{n}\sum_{i=1}^{n}\frac{2z_i\cdot(e^\varepsilon+1)-(e^\varepsilon+1)}{e^\varepsilon-1} \\
&= \frac{1}{n}\sum_{i=1}^{n}(2z_i-1)\frac{(e^\varepsilon+1)}{e^\varepsilon-1} \\
&= \hat{\mu}'.
\end{aligned}$$

We now show that M_{Du} and M_N are equivalent. Recall the definition of the mechanism M_{Du} and its mean estimators (we already recalled the definition of M_N above). The mechanism M_{Du} by Duchi, Jordan, and Wainwright [DJW14; DJW18] (for $d = 1$) takes inputs $x_i \in [-r, r]$ and outputs $z_i \in \{-B, B\}$. Without loss of generality, we assume that $r = 1$.

It calculates $\tilde{x}_i = \begin{cases} +1 & \text{with probability } \frac{1}{2} + \frac{x_i}{2} \\ -1 & \text{with probability } \frac{1}{2} - \frac{x_i}{2}. \end{cases}$ It samples T from a Bernoulli distribution as $T_i \sim \text{Bern}\left(\frac{e^\varepsilon}{e^\varepsilon+1}\right)$. It then calculates $z_i = \begin{cases} B & \text{if } T_i = 1 \text{ and } \tilde{x}_i = 1 \\ -B & \text{if } T_i = 1 \text{ and } \tilde{x}_i = -1 \\ -B & \text{if } T_i = 0 \text{ and } \tilde{x}_i = 1 \\ B & \text{if } T_i = 0 \text{ and } \tilde{x}_i = -1 \end{cases}$ where $B = \frac{e^\varepsilon+1}{e^\varepsilon-1}$.

Note that the possible return values B and $-B$ are the same as the return values of M_N. We therefore only need to show that the sampling distributions of M_{Du} and M_N are the same. We rewrite the output of M_{Du} as

$$\begin{aligned}
Pr[z_i = B] &= Pr[T_i = 1 \text{ and } \tilde{x}_i = 1] + Pr[T_i = 0 \text{ and } \tilde{x}_i = -1] \\
&= Pr[T_i = 1]Pr[\tilde{x}_i = 1] + Pr[T_i = 0]Pr[\tilde{x}_i = -1] \\
&= Pr[T_i = 1]Pr[\tilde{x}_i = 1] + (1 - Pr[T_i = 1])(1 - Pr[\tilde{x}_i = 1]) \\
&= Pr[T_i = 1]Pr[\tilde{x}_i = 1] + 1 - Pr[\tilde{x}_i = 1] \\
&\quad - Pr[T_i = 1] + Pr[T_i = 1]Pr[\tilde{x}_i = 1] \\
&= 2Pr[T_i = 1]Pr[\tilde{x}_i = 1] + 1 - Pr[\tilde{x}_i = 1] - Pr[T_i = 1] \\
&= 2\frac{e^\varepsilon}{e^\varepsilon + 1}\left(\frac{1}{2} + \frac{x_i}{2}\right) + 1 - \left(\frac{1}{2} + \frac{x_i}{2}\right) - \frac{e^\varepsilon}{e^\varepsilon + 1} \\
&= \frac{e^\varepsilon}{e^\varepsilon + 1} + \frac{e^\varepsilon}{e^\varepsilon + 1}x_i + \frac{1}{2} - \frac{x_i}{2} - \frac{e^\varepsilon}{e^\varepsilon + 1} \\
&= \frac{e^\varepsilon}{e^\varepsilon + 1}x_i + \frac{e^\varepsilon + 1}{e^\varepsilon + 1}(\frac{1}{2} - \frac{x_i}{2}) \\
&= \frac{1}{2e^\varepsilon + 2}\left(2e^\varepsilon x_i + e^\varepsilon - e^\varepsilon x_i + 1 - x_i\right) \\
&= \frac{(e^\varepsilon - 1)x_i + e^\varepsilon + 1}{2e^\varepsilon + 2}
\end{aligned}$$

This is the same as the probability p' of M_N. Therefore, the mechanisms M_{Du} and M_N are equivalent. □

Proof of Proposition 2. We assume that the error of the mean estimator is bounded by $f(n, \varepsilon)$, i.e., $|\hat{\mu} - \mu| \leq f(n, \varepsilon)$. We now derive an upper bound for the error of the variance estimator.

$$
\begin{aligned}
\left|\hat{s}_X^2 - s_X^2\right| =& \left|\frac{n}{n-1}(\hat{\mu}_{X^2} - \hat{\mu}_X^2) - \frac{n}{n-1}(\mu_{X^2} - \mu_X^2)\right| \\
=& \frac{n}{n-1}\left|\hat{\mu}_{X^2} - \mu_{X^2} - \hat{\mu}_X^2 + \mu_X^2\right| \\
=& \frac{n}{n-1}\left|(\hat{\mu}_{X^2} - \mu_{X^2}) + (-(\hat{\mu}_X^2 - \mu_X^2))\right| \\
& \textit{(applying the Triangle Inequality)} \\
\leq& \frac{n}{n-1}(|\hat{\mu}_{X^2} - \mu_{X^2}| + \left|\hat{\mu}_X^2 - \mu_X^2\right|) \\
& \textit{(using } |\hat{\mu} - \mu| \leq f(n, \varepsilon)\textit{)} \\
\leq& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + |(\hat{\mu}_X + \mu_X)(\hat{\mu}_X - \mu_X)|\right) \\
=& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + |\hat{\mu}_X + \mu_X|\,|\hat{\mu}_X - \mu_X|\right) \\
& \textit{(using } |\hat{\mu} - \mu| \leq f(n, \varepsilon)\textit{)} \\
\leq& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + |\hat{\mu}_X + \mu_X|\, f(n_X, \varepsilon_X)\right) \\
=& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + |\hat{\mu}_X - \mu_X + \mu_X + \mu_X|\, f(n_X, \varepsilon_X)\right) \\
& \textit{(applying the Triangle Inequality)} \\
\leq& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + (|\hat{\mu}_X - \mu_X| + |\mu_X + \mu_X|)\, f(n_X, \varepsilon_X)\right) \\
& \textit{(using } |\hat{\mu} - \mu| \leq f(n, \varepsilon) \textit{ and } \mu_X \leq 1 \textit{ if } x_i \in [-1, 1]\textit{)} \\
\leq& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + (f(n_X, \varepsilon_X) + 2)\, f(n_X, \varepsilon_X)\right) \\
=& \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon_{X^2}) + f(n_X, \varepsilon_X)^2 + 2f(n_X, \varepsilon_X)\right)
\end{aligned}
$$

In the case where we split $\varepsilon = \varepsilon_X + \varepsilon_{X^2}$, we have $n = n_X = n_{X^2}$. Therefore, we have

$$
\left|\hat{s}_X^2 - s_X^2\right| \leq \frac{n}{n-1}\left(f(n, \varepsilon_{X^2}) + f(n, \varepsilon_X)^2 + 2f(n, \varepsilon_X)\right)
$$

In the case where we split $n = n_X + n_{X^2}$, we use the same ε for all participants: $\varepsilon = \varepsilon_X = \varepsilon_{X^2}$. Therefore, we have

$$
\left|\hat{s}_X^2 - s_X^2\right| \leq \frac{n}{n-1}\left(f(n_{X^2}, \varepsilon) + f(n_X, \varepsilon)^2 + 2f(n_X, \varepsilon)\right)
$$

□

C Error Bound of the Mean Estimator by Waudby-Smith et al. (2023)

Waudby-Smith, Wu, and Ramdas [WWR23] produce a confidence interval that contains the population mean μ with probability $1 - \alpha$.

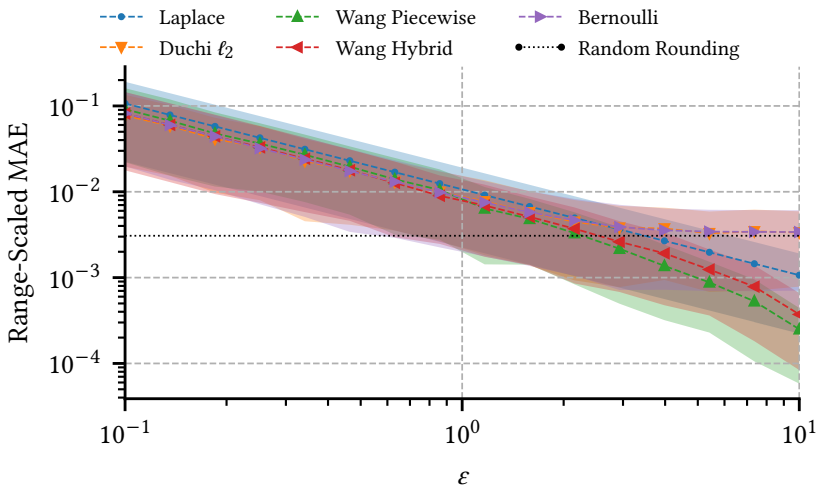

Figure 5: Mean absolute error of the mean estimation averaged over all datasets (scaled by the respective input range) and $n = 10^4$. Shaded areas indicate the standard deviation. "Duchi" and "Wang" refer to the works by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18] and Wang et al. [Wan+19a] respectively. Ding, Kulkarni, and Yekhanin [DKY17], Nguyên et al. [Ngu+16], and Waudby-Smith, Wu, and Ramdas [WWR23], and the ℓ_∞ mechanism by Duchi, Wainwright, and Jordan [DWJ13] are equivalent and are summarized as "Bernoulli Mechanisms".

The lower and upper bounds are defined as $\hat{\mu}_n \pm \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{1}{n}\sum_{i=1}^n r_i\right)^2}}$. Since we set $\varepsilon_i = \varepsilon$ for all clients and follow the authors' recommendation $G_i = 1$, $r_i = \frac{e^\varepsilon - 1}{e^\varepsilon + 1}$. Therefore, the bounds are $\hat{\mu}_n \pm \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{e^\varepsilon-1}{e^\varepsilon+1}\right)^2}}$. Since the population mean μ is within the bounds with probability $1 - \alpha$,

$$\hat{\mu}_n - \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{e^\varepsilon-1}{e^\varepsilon+1}\right)^2}} \leq \mu \leq \hat{\mu}_n + \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{e^\varepsilon-1}{e^\varepsilon+1}\right)^2}} \tag{2}$$

Therefore, we have with probability $1 - \alpha$:

$$|\hat{\mu}_n - \mu| \leq \left|\hat{\mu}_n - \hat{\mu}_n - \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{e^\varepsilon-1}{e^\varepsilon+1}\right)^2}}\right| \tag{3}$$

$$= \sqrt{\frac{\log(1/\alpha)}{2n\left(\frac{e^\varepsilon-1}{e^\varepsilon+1}\right)^2}} \tag{4}$$

$$= \frac{1}{\sqrt{2n}} \frac{e^\varepsilon + 1}{e^\varepsilon - 1} \sqrt{\log(1/\alpha)} \tag{5}$$

D Additional Figures

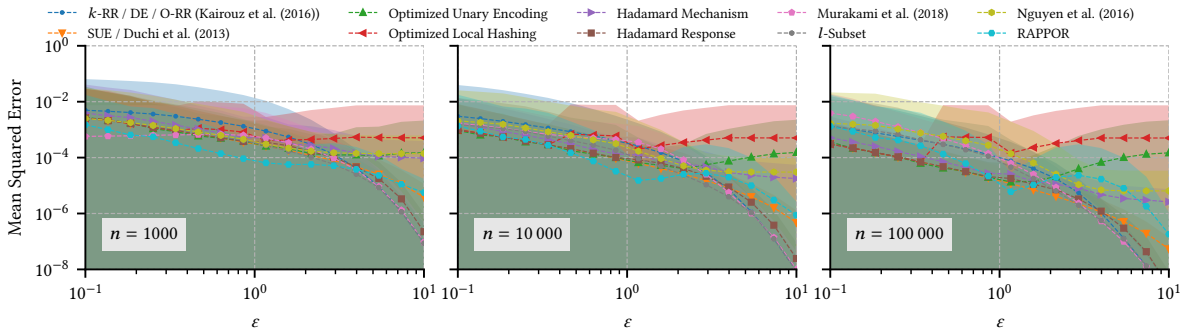

Figure 6: Mean squared error of the frequency estimation averaged over all datasets with $n = 10^3, n = 10^4, n = 10^5$. Shaded areas indicate the standard deviation.

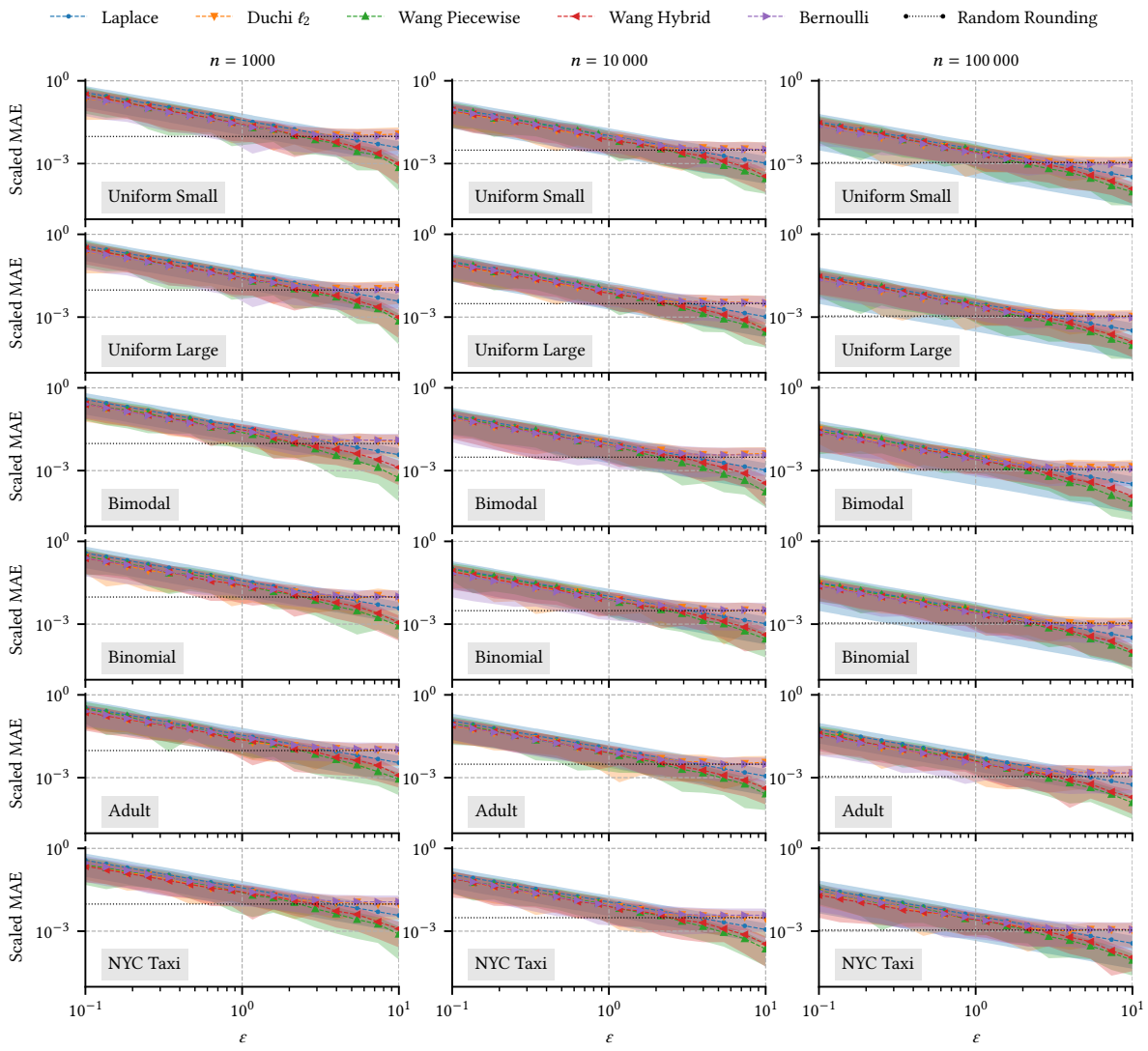

Figure 7: Mean absolute error for 1-dimensional mean estimation. Shaded areas indicate the standard deviation. All results are averaged over 100 runs. "Duchi ℓ_2" refers to the ℓ_2 method by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18], "Wang" refers to the method by Wang et al. [Wan+19a], and "Bernoulli" groups the methods by Ding, Kulkarni, and Yekhanin [DKY17], Nguyên et al. [Ngu+16], Waudby-Smith, Wu, and Ramdas [WWR23], and the ℓ_∞ method by Duchi, Wainwright, and Jordan [DWJ13].

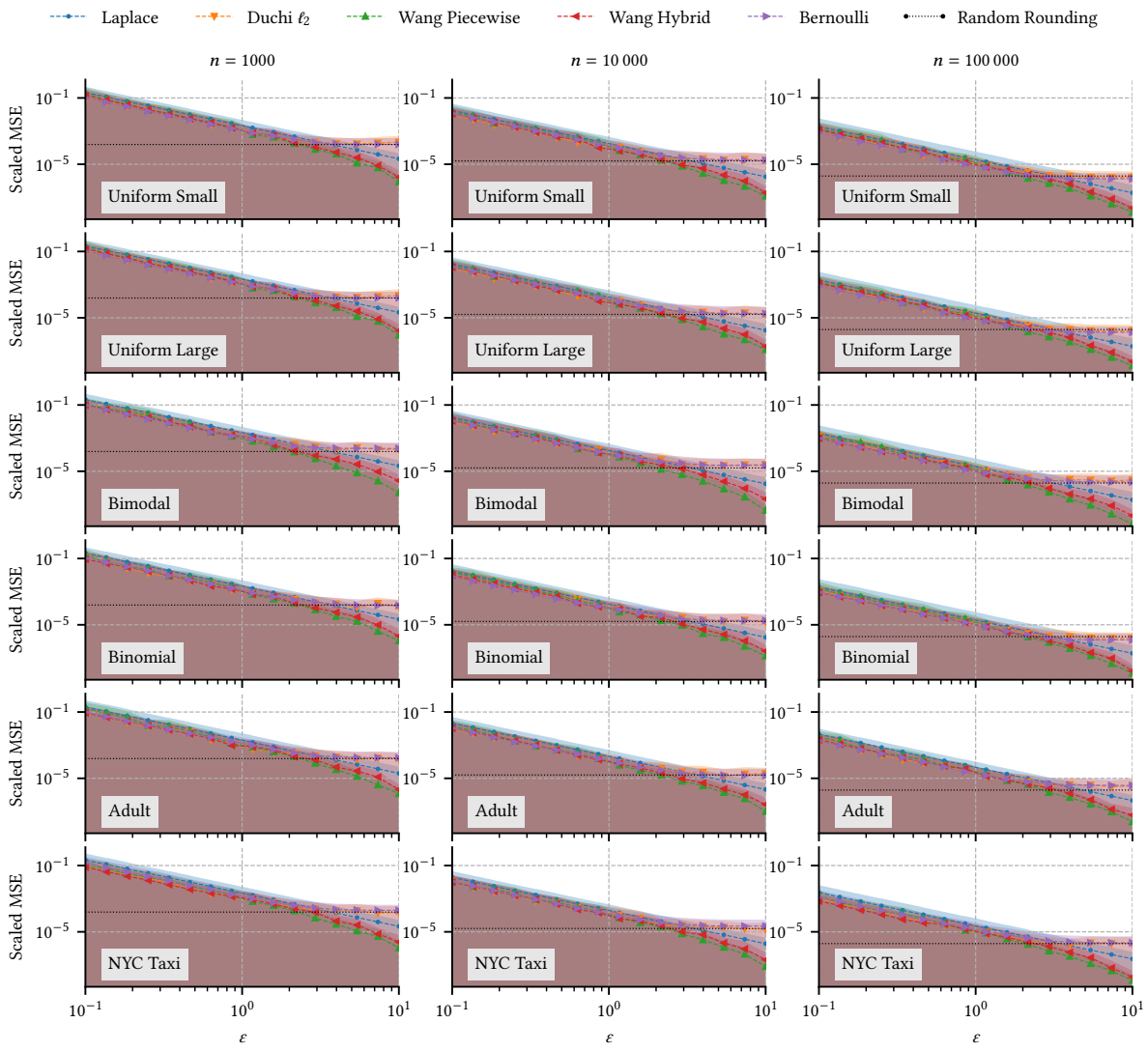

Figure 8: Mean squared error for 1-dimensional mean estimation. Shaded areas indicate the standard deviation. All results are averaged over 100 runs. "Duchi ℓ_2" refers to the ℓ_2 method by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18], "Wang" refers to the method by Wang et al. [Wan+19a], and "Bernoulli" groups the methods by Ding, Kulkarni, and Yekhanin [DKY17], Nguyên et al. [Ngu+16], Waudby-Smith, Wu, and Ramdas [WWR23], and the ℓ_∞ method by Duchi, Wainwright, and Jordan [DWJ13].

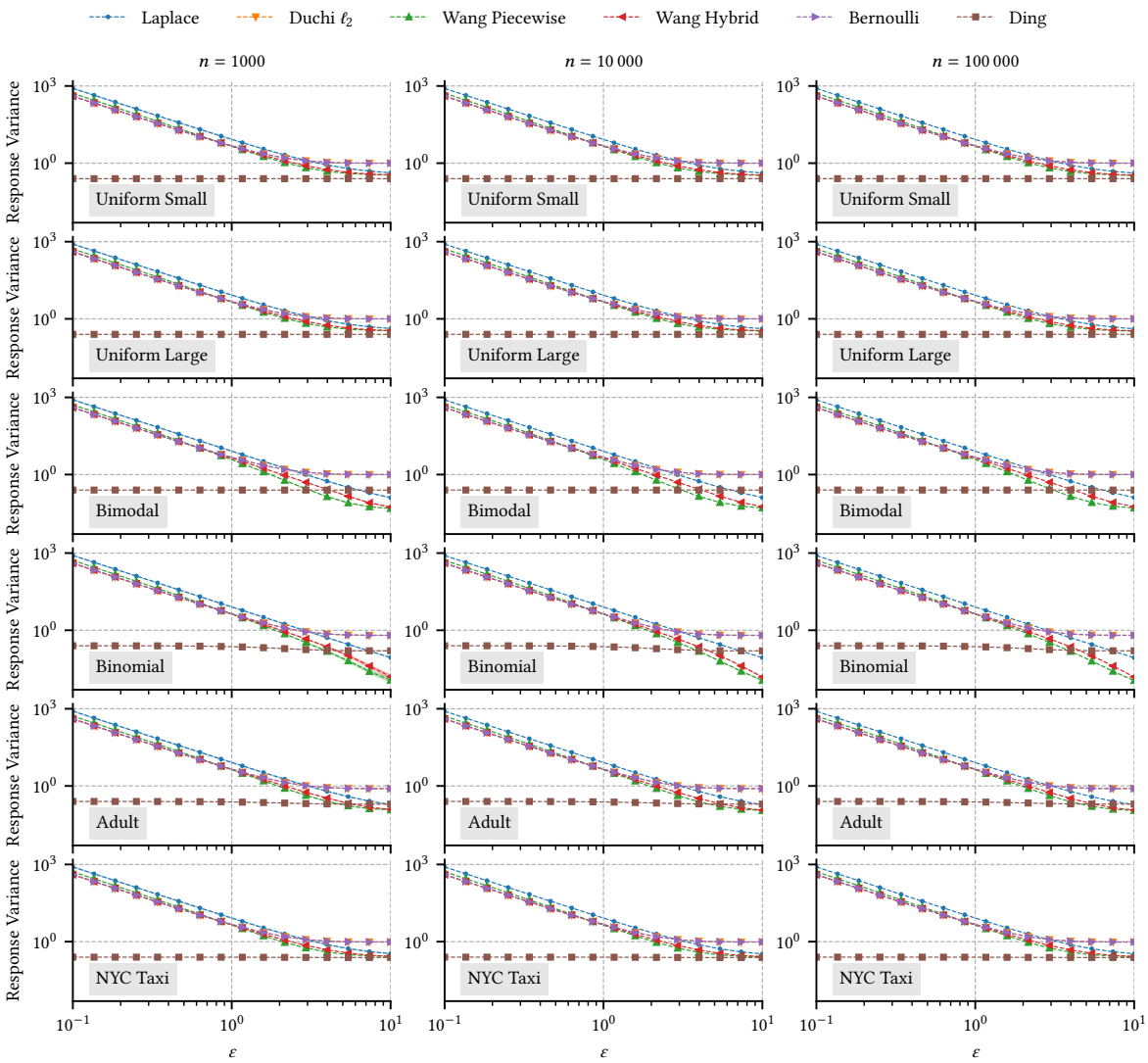

Figure 9: Variance of the participants' responses for 1-dimensional mean estimation. Shaded areas indicate the standard deviation. All results are averaged over 100 runs. "Duchi ℓ_2" refers to the ℓ_2 method by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18], "Wang" refers to the method by Wang et al. [Wan+19a], and "Bernoulli" groups the methods by Nguyên et al. [Ngu+16], Waudby-Smith, Wu, and Ramdas [WWR23], and the ℓ_∞ method by Duchi, Wainwright, and Jordan [DWJ13]. The variance of the responses generated by the method by Ding, Kulkarni, and Yekhanin [DKY17] is not included in the "Bernoulli" group here as its responses are biased and therefore show a different behavior.

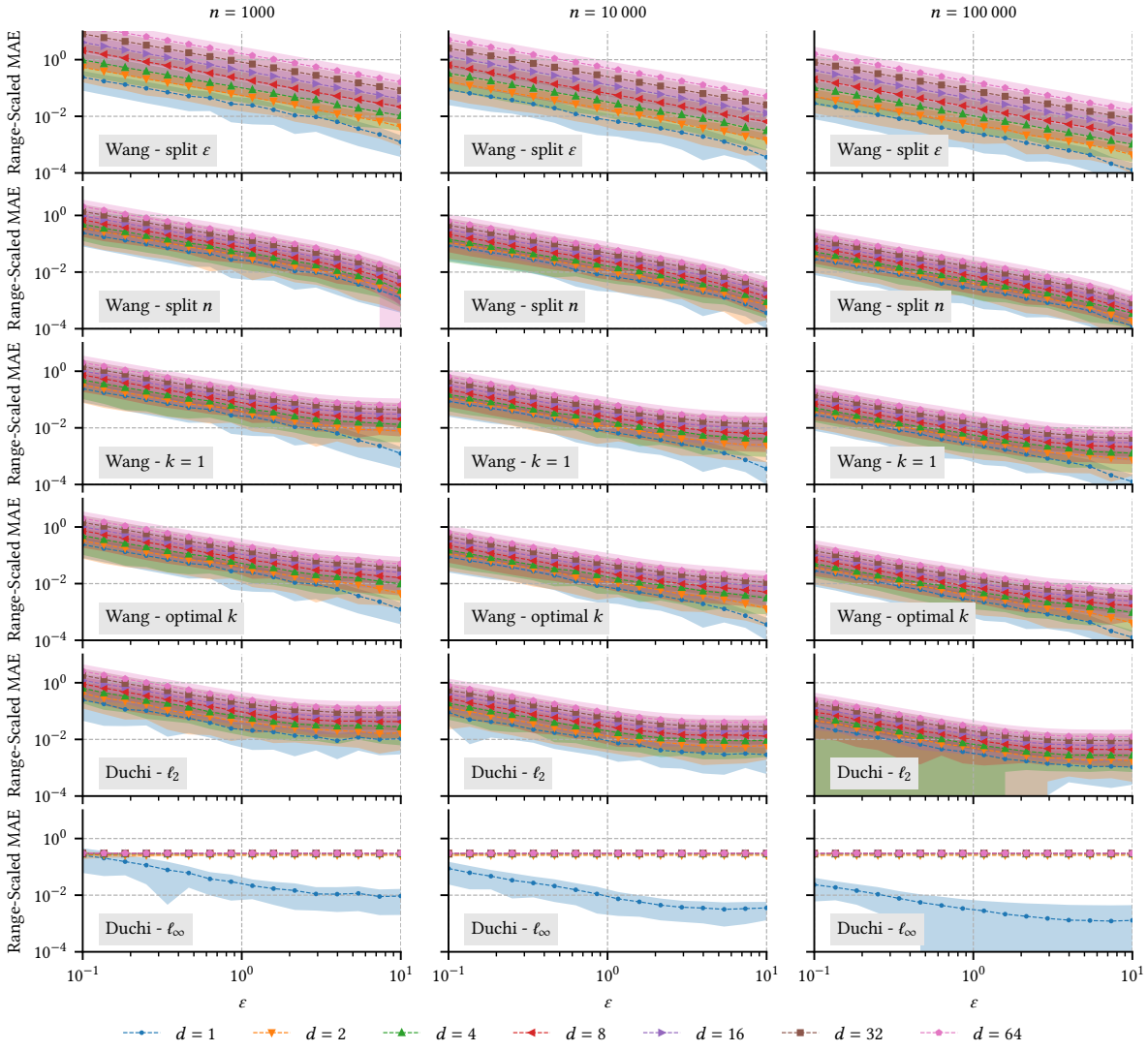

Figure 10: Mean absolute error for multi-dimensional mean estimation. Shaded areas indicate the standard deviation. All results are averaged over 100 runs. "Wang" refers to the method by Wang et al. [Wan+19a] with "optimal k" using the value suggested by the authors. "Wang k=1" refers to the same method with $k = 1$ (as in the earlier version of the paper by Nguyên et al. [Ngu+16]). The variants "split ε" and "split n" refer to the naive approaches of splitting the privacy budget or the participants by the number of dimensions and use the method by Wang et al. [Wan+19a] for the 1-dimensional mean estimation. "Duchi" refers to the methods by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18].

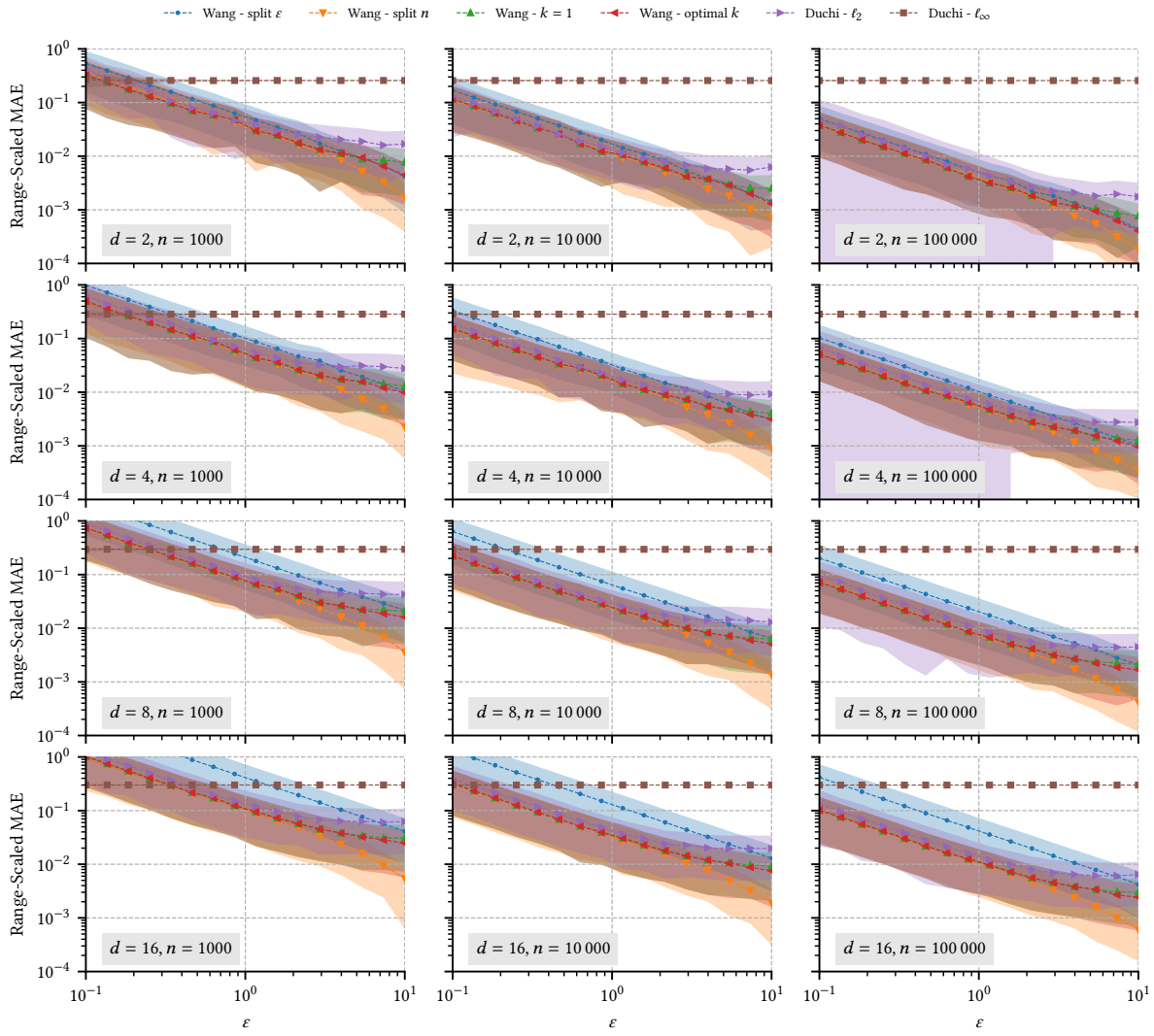

Figure 11: Mean absolute error for multi-dimensional mean estimation. Shaded areas indicate the standard deviation. All results are averaged over 100 runs. "Wang" refers to the method by Wang et al. [Wan+19a] with "optimal k" using the value suggested by the authors. "Wang k=1" refers to the same method with $k = 1$ (as in the earlier version of the paper by Nguyên et al. [Ngu+16]). The variants "split ε" and "split n" refer to the naive approaches of splitting the privacy budget or the participants by the number of dimensions and use the method by Wang et al. [Wan+19a] for the 1-dimensional mean estimation. "Duchi" refers to the methods by Duchi, Wainwright, and Jordan [DWJ13] and Duchi, Jordan, and Wainwright [DJW18].

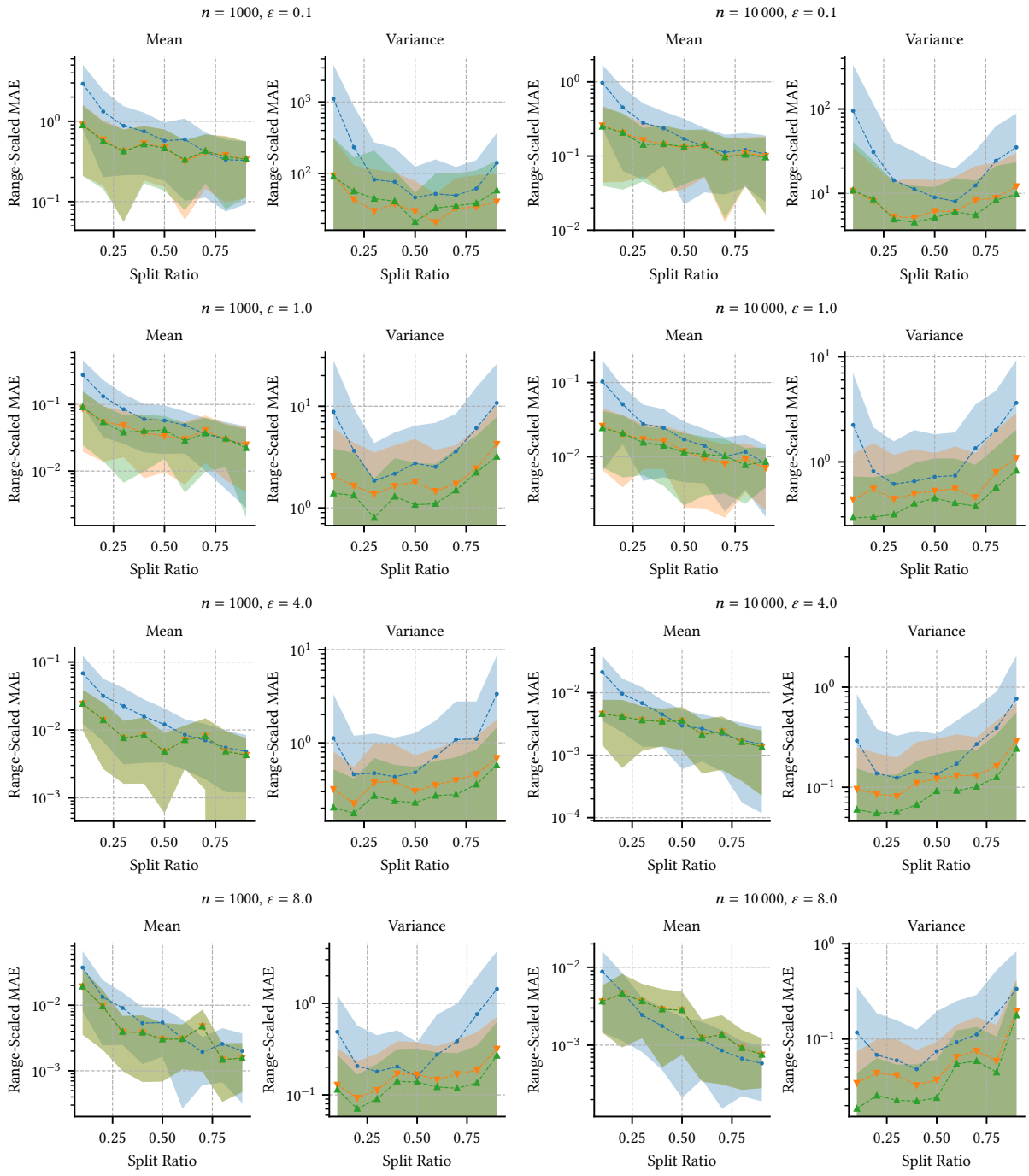

Figure 12: Mean absolute error of the variance estimation averaged over all datasets (scaled by the respective input range) with $n = \{10^3, 10^4\}$ and $\varepsilon = \{0.1, 1, 4, 8\}$. The split ratio defines how much of the privacy budget (or participants) is used for the mean estimation step. The variance estimation uses the Piecewise mechanism by Wang et al. [Wan+19a] as the underlying mean estimator.

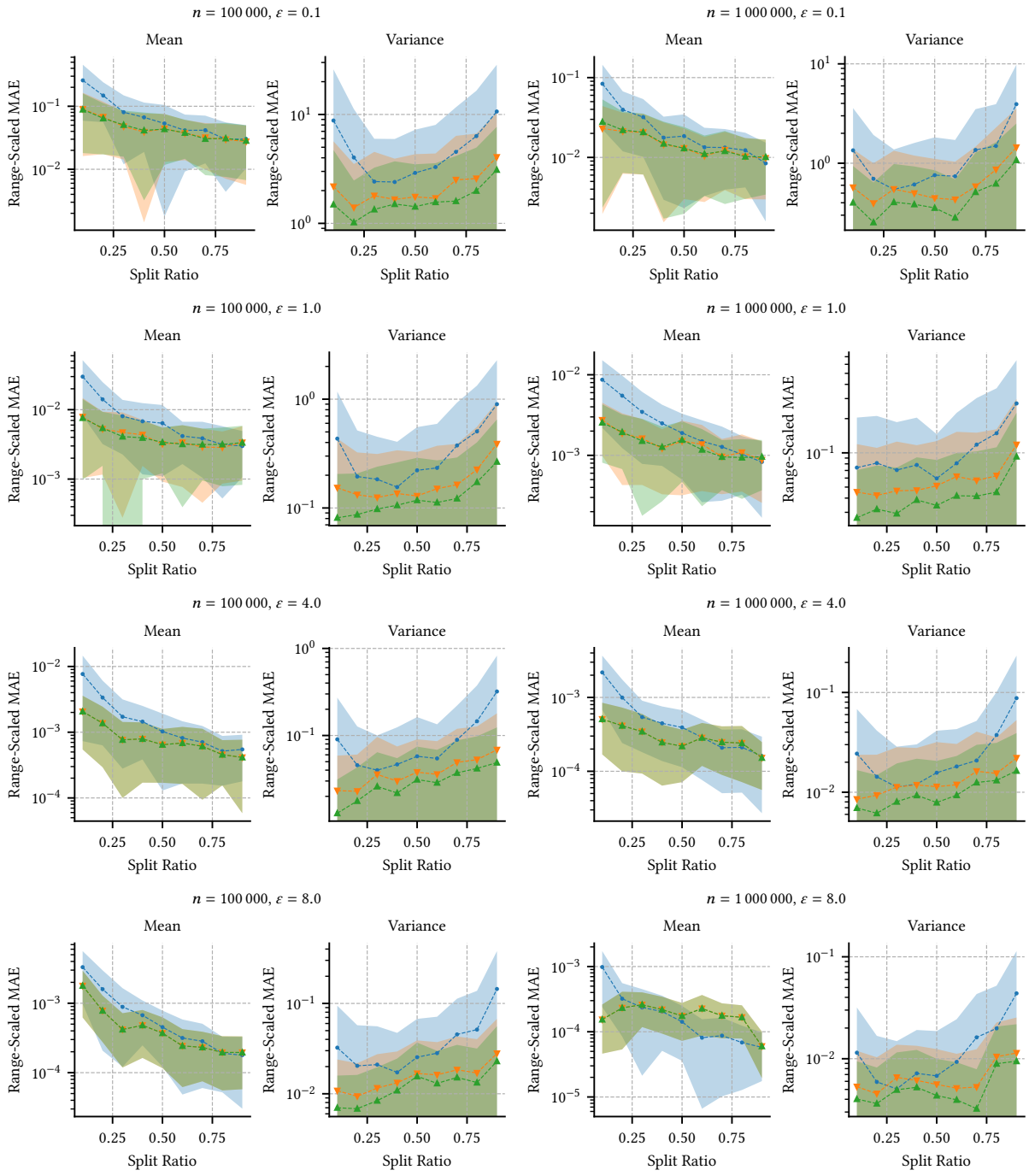

Figure 13: Mean absolute error of the variance estimation averaged over all datasets (scaled by the respective input range) with $n = \{10^5, 10^6\}$ and $\varepsilon = \{0.1, 1, 4, 8\}$. The split ratio defines how much of the privacy budget (or participants) is used for the mean estimation step. The variance estimation uses the Piecewise mechanism by Wang et al. [Wan+19a] as the underlying mean estimator.

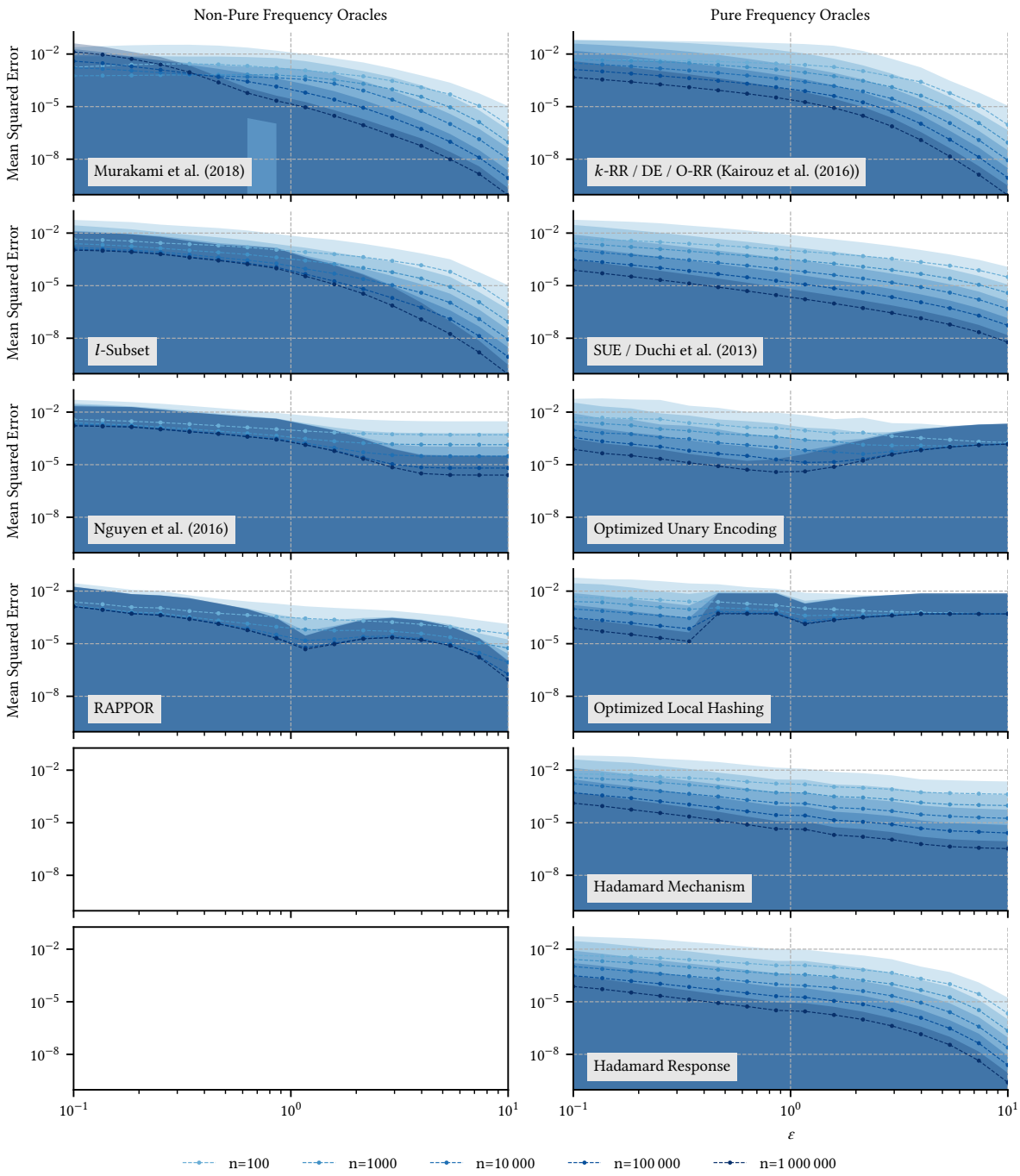

Figure 14: Mean squared error for $n = \{10^2, 10^3, 10^4, 10^5, 10^6, 10^7\}$ for pure and non-pure frequency oracles averaged over all datasets (see table 3). Shaded areas indicate the standard deviation. All results are averaged over 20 runs and each estimate is post-processed by projection onto the probability simplex.

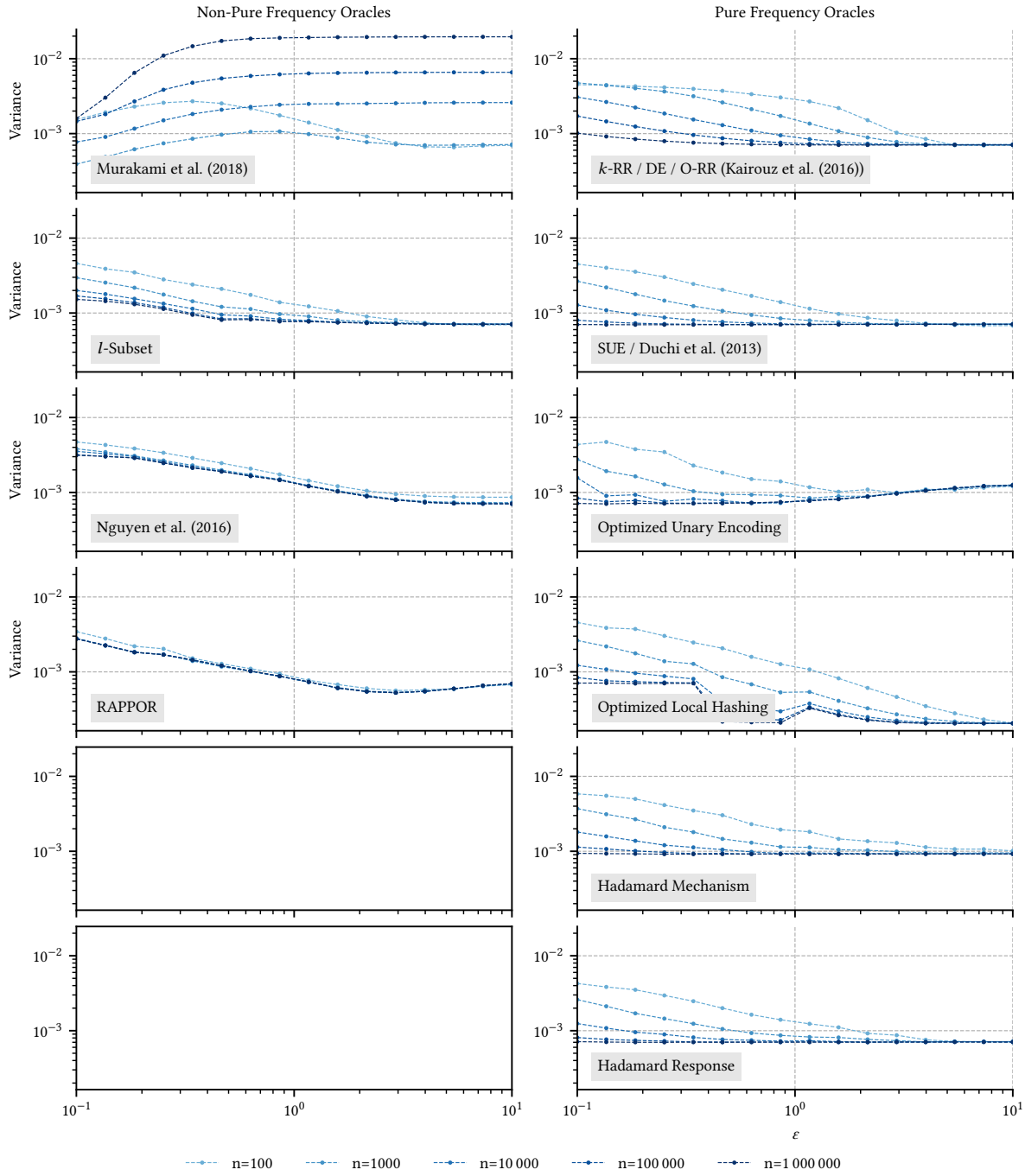

Figure 15: Variance of the estimated frequencies for $n = \{10^2, 10^3, 10^4, 10^5, 10^6, 10^7\}$ for pure and non-pure frequency oracles averaged over all datasets (see table 3). All results are averaged over 20 runs and each estimate is post-processed by projection onto the probability simplex.

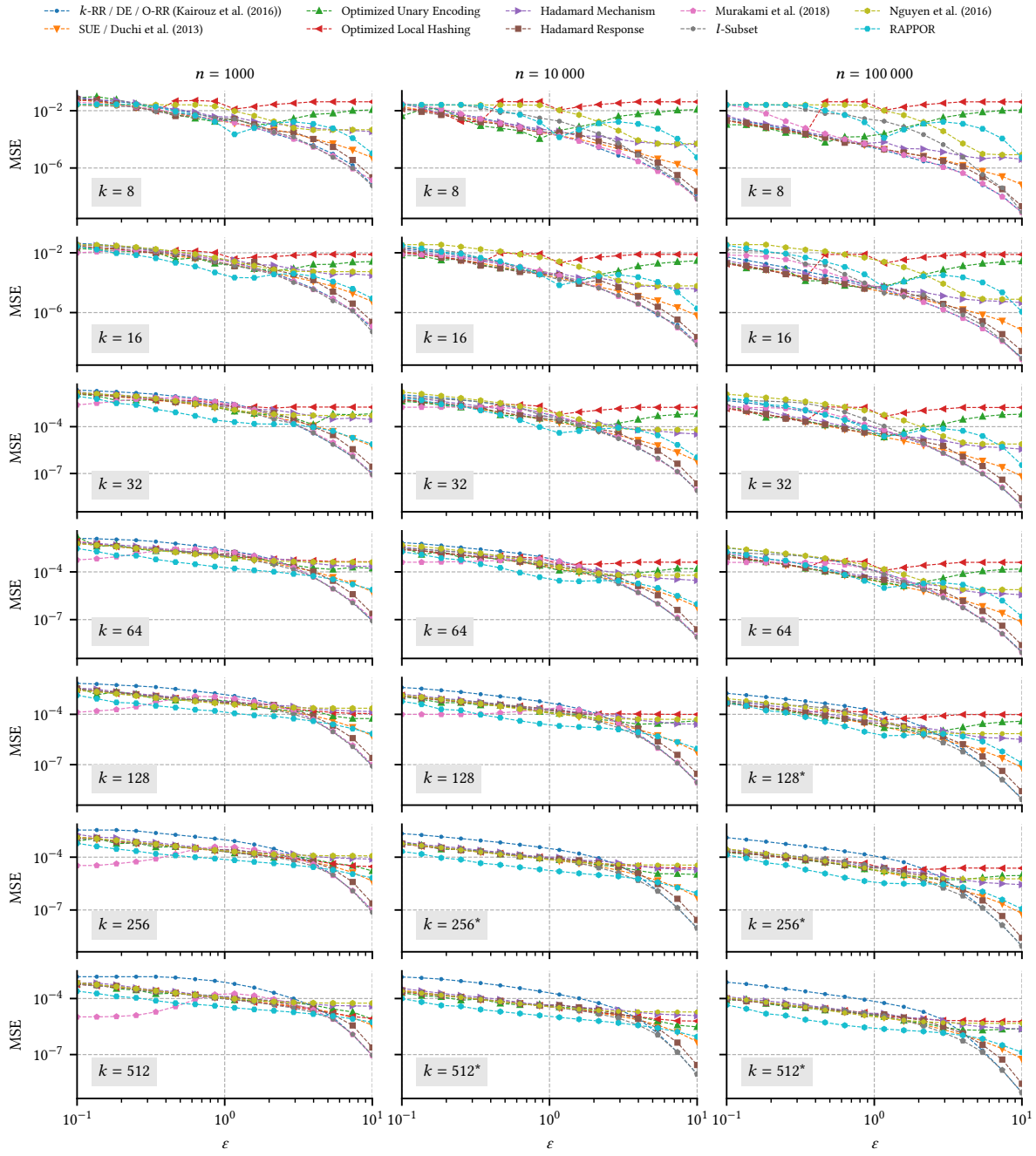

Figure 16: Mean squared error for $n = \{10^3, 10^4, 10^5\}$ and domain size $k = \{8, 16, 32, 64, 128, 256, 512\}$ for pure and non-pure frequency oracles for the geometric dataset (see table 3). Note that the results for Murakami, Hino, and Sakuma [MHS18] are missing in the subplots marked with an asterisk due to the computational overhead. All results are averaged over 20 runs and each estimate is post-processed by projection onto the probability simplex.

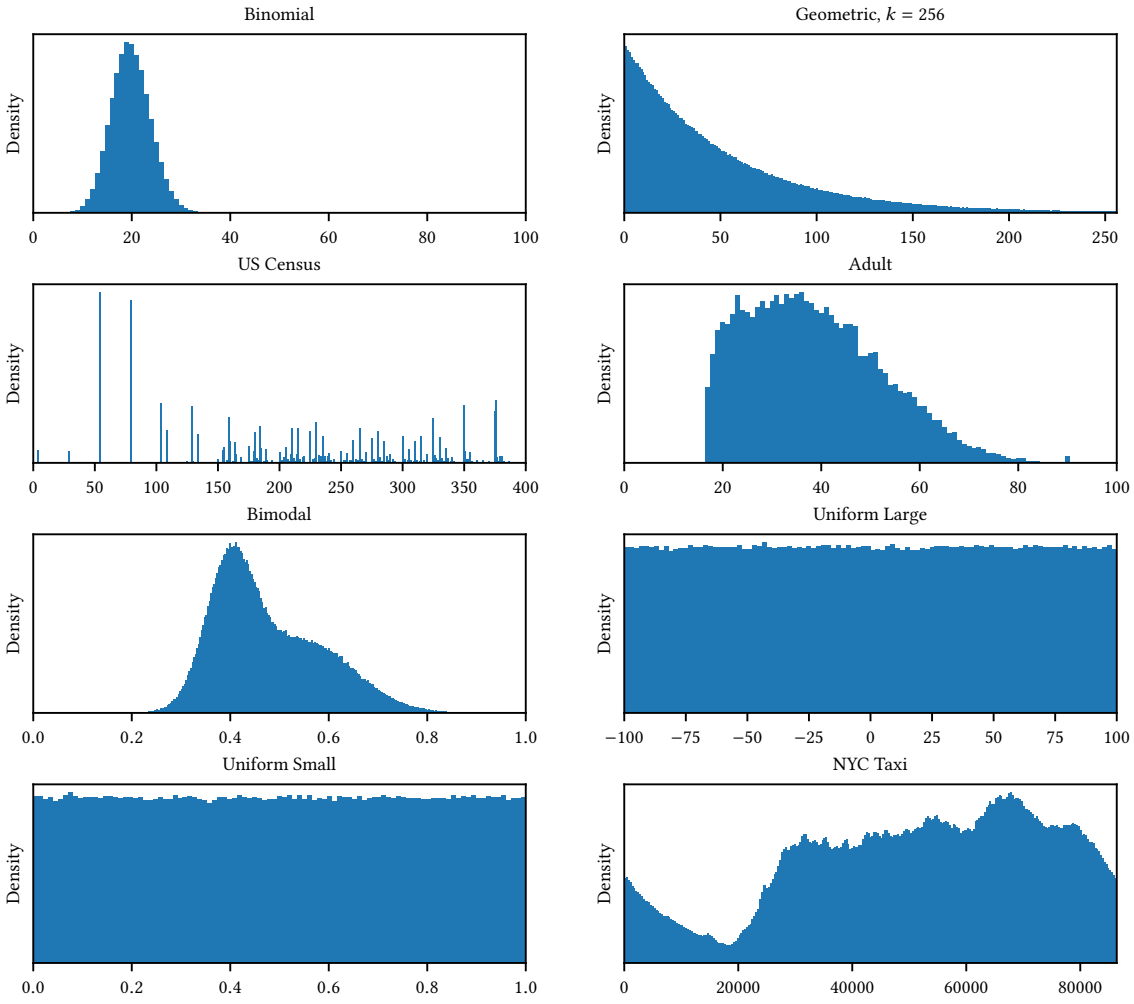

Figure 17: Visualization of the datasets used in the empirical evaluation (see Table 3).