Leveraging Public Data for Practical Private Query Release

Terrance Liu [1] **Giuseppe Vietri** [2] **Thomas Steinke** [3] **Jonathan Ullman** [4] **Zhiwei Steven Wu** [1]

Abstract

In many statistical problems, incorporating priors can significantly improve performance. However, the use of prior knowledge in differentially private query release has remained underexplored, despite such priors commonly being available in the form of public datasets, such as previous US Census releases. With the goal of releasing statistics about a private dataset, we present $\mathsf{PMW}^{\mathsf{Pub}}$, which—unlike existing baselines—leverages public data drawn from a related distribution as prior information. We provide a theoretical analysis and an empirical evaluation on the American Community Survey (ACS) and ADULT datasets, which shows that our method outperforms state-of-the-art methods. Furthermore, $\mathsf{PMW}^{\mathsf{Pub}}$ scales well to high-dimensional data domains, where running many existing methods would be computationally infeasible.

1. Introduction

As the collection and distribution of private information becomes more prevalent, controlling privacy risks is becoming a priority for organizations that depend on this data. Differential privacy (Dwork et al., 2006b) is a rigorous criterion that provides meaningful guarantees of individual privacy while allowing for trade-offs between privacy and accuracy. It is now deployed by organizations such as Google, Apple, and the US Census Bureau. In this work, we study the problem of *differentially private query release*, specifically generating a *private synthetic dataset*: a new dataset in which records are "fake" but the statistical properties of the original data are preserved. In particular, the release of summary data from the 2020 US Decennial Census—one of the most notable applications of differential privacy (Abowd, 2018)—can be framed as a private query release problem.

In practice, generating accurate differentially private synthetic datasets is challenging without an excessively large private dataset, and a promising avenue for improving these algorithms is to find methods for incorporating *prior information* that lessen the burden on the private data. In this paper we explore using public data as one promising source of prior information that can be used without regard for its privacy.[1] For example, one can derive auxiliary data for the 2020 US Census release from already-public releases like the 2010 US Census. Similarly, the Census Bureau's American Community Survey has years of annual releases that can be treated as public data for future releases. Alternatively, once national-level statistics are computed and released, they can serve as public data for computing private statistics over geographic subdivisions, such as states and counties. Indeed, such a hierarchy of releases is part of the $\mathsf{TopDown}$ algorithm being developed for the 2020 US Census (Abowd et al., 2019).

Existing algorithms for private query release do not incorporate public data. While there is theoretical work on *public-data-assisted private query release* (Bassily et al., 2020a), it crucially assumes that the public and private data come from the same distribution and does not give efficient algorithms.

Our Contributions Therefore, in light of these observations, we make the following contributions:

1. We initiate the study of using public data to improve private query release in the more realistic setting where the public data is from a distribution that is *related but not identical* to the distribution of the private data.

2. We present (Private) Multiplicative Weights with Public Data ($\mathsf{PMW}^{\mathsf{Pub}}$), an extension of MWEM (Hardt et al., 2012) that incorporates public data.

3. We show that as a side benefit of leveraging public data, $\mathsf{PMW}^{\mathsf{Pub}}$ is computationally efficient and therefore is practical for much larger problem sizes than MWEM.

4. We analyze the theoretical privacy and accuracy guarantees of $\mathsf{PMW}^{\mathsf{Pub}}$.

[1] Carnegie Mellon University, Pittsburgh, PA, USA [2] University of Minnesota, Minneapolis, MN, USA [3] Google, Mountain View, CA, USA [4] Northeastern University, Boston, MA, USA. Correspondence to: Zhiwei Steven Wu <zstevenwu@cmu.edu>.

Proceedings of the 38th International Conference on Machine Learning, PMLR 139, 2021. Copyright 2021 by the author(s).

[1] The public data may have been derived from private data, but we refer to it as "public" for our purposes as long as the privacy concerns have already been addressed.

5. We empirically evaluate $\mathsf{PMW}^{\mathsf{Pub}}$ on the American Community Survey (ACS) data to demonstrate that we can achieve strong performance when incorporating public data, even when public samples come from a different distribution.

1.1. Related Work

Our work relates to a growing line of research that utilizes publicly available data for private data analyses in the setting where the public and private data come from the same distribution. For private query release, Bassily et al. (2020a) prove upper and lower bounds on the number of public and private samples needed, and Alon et al. (2019) do the same for binary classification. Neither of these works, however, give computationally efficient algorithms. Other works consider a more general problem, *private prediction*, where the public data is unlabeled and the private data is used to label the public data (Bassily et al., 2018; Dwork & Feldman, 2018; Nandi & Bassily, 2020; Papernot et al., 2018). Beimel et al. (2013) consider the *semi-private* setting where only a portion of examples require privacy for the labels.

These prior works are limited by the strong assumption that the public and private data are drawn from the same distribution. One notable exception is the recent work of Bassily et al. (2020b) on supervised learning, in which the authors assume that the public and private data can be labeled differently but have the same marginal distribution without labels. Given that their problem is trivial otherwise, they focus solely on the setting where the public dataset is smaller than the private dataset. However, if the public data comes from a different distribution (as is the case in our experiments), the setting in which the size of the public dataset is similar to or larger than that of the private dataset becomes interesting.

Finally, Ji & Elkan (2013) propose a method that, like $\mathsf{PMW}^{\mathsf{Pub}}$, reweights a support derived from a public dataset (via importance weighting). However, while their method does not rely on the assumption that the public and private data come from the same distribution, Ji & Elkan (2013) do not make this distinction in their theoretical analysis or discussion. Moreover, unlike the algorithm presented in this work, their method is not tailored to the problem of query release.[2]

2. Preliminaries

We consider a data domain $\mathcal{X} = \{0,1\}^d$ of dimension d, a private dataset $\widetilde{D} \in \mathcal{X}^n$ consisting of the data belonging to n individuals, and a class of statistical linear queries $\mathcal{Q}$. Our final objective is to generate a synthetic dataset in a privacy-preserving way that matches the private data's answers. Consider a randomized mechanism $\mathcal{M} : \mathcal{X}^n \to \mathcal{R}$ that takes as input a private dataset $\widetilde{D}$ and computes a synthetic dataset $X \in \mathcal{R}$, where $\mathcal{R}$ represents the space of possible datasets. Given a set of queries $\mathcal{Q}$, we say that the max error of a synthetic dataset X is given by $\max_{q \in \mathcal{Q}} |q(\widetilde{D}) - q(X)|$.

We begin with the definition of a statistical linear query:

Definition 2.1 (Statistical linear query)**.** Given a predicate ϕ and a dataset D, the linear query $q_\phi : \mathcal{X}^n \to [0,1]$ is defined by

$$q_\phi(D) = \frac{1}{|D|} \sum_{x \in D} \phi(x)$$

Defining a dataset instead as a distribution A over the domain $\mathcal{X}$, the definition for a linear query q_ϕ then becomes $q_\phi(A) = \sum_{x \in \mathcal{X}} \phi(x) A(x)$.

One example of a statistical query class is k-way marginal queries, which we define below.

Definition 2.2 (k-way marginal query)**.** Let the data universe with d categorical attributes be $\mathcal{X} = (\mathcal{X}_1 \times \ldots \times \mathcal{X}_d)$, where each $\mathcal{X}_i$ is the discrete domain of the ith attribute. A k-way marginal query $\phi_{S,y}$ is a linear query specified by a set of k attributes $S \subseteq [d]$ (with $|S| = k$) and target $y \in \prod_{i \in S} \mathcal{X}_i$ such that for all $x \in \mathcal{X}$

$$\phi_{S,y}(x) = \begin{cases} 1 & : \forall j \in S \quad x_j = y_j \\ 0 & : \text{otherwise} \end{cases}$$

where $x_i \in \mathcal{X}_i$ means the ith attribute of record $x \in \mathcal{X}$. We define a *workload* as the set of marginal queries given by a set of attributes S. The workload given by attributes in S has a total of $\prod_{i \in S} |\mathcal{X}_i|$ marginal queries.

Although we evaluate on k-way marginal queries in our experiments, we provide theoretical results that hold for any class of linear queries.

Definition 2.3 (ℓ_1-sensitivity)**.** The ℓ_1-sensitivity of a function $f : \mathcal{X}^* \to \mathbb{R}^k$ is

$$\Delta f = \max_{\text{neighboring } D, D'} \|f(D) - f(D')\|_1$$

In the context of statistical queries, the ℓ_1-sensitivity of query captures the effect of changing an individual in the dataset and is useful for determining the amount of perturbation required for preserving privacy.

In our setting, we have access to a public dataset $\widehat{D} \in \mathcal{X}^m$ containing the data of m individuals that we can use without privacy constraints. This dataset defines a public data domain, denoted by $\widehat{\mathcal{X}} \subset \mathcal{X}$, which consists of all unique rows in $\widehat{D}$. We assume that both the public and private datasets are i.i.d. samples from different distributions and use the Rényi divergence, which we define below, as a measure for how close the two distributions are.

[2] See Appendix A.5 for an additional discussion.

Definition 2.4 (Rényi divergence)**.** Let μ and ν be probability distributions on Ω. For $\alpha \in (1, \infty)$, we define the Rényi divergence of order α between μ and ν as

$$\mathrm{D}_\alpha(\mu \parallel \nu) = \frac{1}{1-\alpha} \log \sum_{x \in \Omega} \mu(x)^\alpha \nu(x)^{1-\alpha}$$

The Rényi divergence is also used in the definition of privacy that we adopt. The output of a randomized mechanism $\mathcal{M} : \mathcal{X}^* \to \mathcal{R}$ is a privacy preserving-computation if it satisfies concentrated differential privacy (CDP) (Dwork & Rothblum, 2016; Bun & Steinke, 2016):

Definition 2.5 (Concentrated DP)**.** A randomized mechanism $M : \mathcal{X}^n \to \mathcal{R}$ is $\frac{1}{2}\tilde{\varepsilon}^2$-CDP, if for all neighboring datasets D, D' (i.e., differing on a single person), and for all $\alpha \in (1, \infty)$,

$$\mathrm{D}_\alpha(\mathcal{M}(D) \parallel \mathcal{M}(D')) \leq \frac{1}{2}\tilde{\varepsilon}^2\alpha$$

where $\mathrm{D}_\alpha(\mathcal{M}(D) \parallel \mathcal{M}(D'))$ is the Rényi divergence between the distributions of $\mathcal{M}(D)$ and $\mathcal{M}(D')$.

Two datasets are *neighboring* if you can obtain one from the other by changing the data of one individual. Definition 2.5 says that a randomized mechanism computing on a dataset satisfies zCDP if its output distribution does not change by much in terms of Rényi divergence when a single user in the dataset is changed. Finally, any algorithm that satisfies zCDP also satisfies (approximate) differential privacy (Dwork et al., 2006b;a):

Definition 2.6 (Differential Privacy (DP))**.** A randomized algorithm $\mathcal{M} : \mathcal{X}^* \to \mathcal{R}$ satisfies (ε, δ)-differential privacy (DP) if for all neighboring databases D, D', and every event $E \subseteq \mathcal{R}$, we have

$$\Pr[\mathcal{M}(D) \in E] \leq e^\varepsilon \Pr[\mathcal{M}(D') \in E] + \delta.$$

If $\delta = 0$, we say that $\mathcal{M}$ satisfies pure (or pointwise) ε-differential privacy.

3. Public Data Assisted MWEM

In this section, we revisit MWEM and then introduce PMW$^{\mathsf{Pub}}$, which adapts MWEM to leverage public data.

3.1. MWEM

MWEM (Hardt et al., 2012) is an approach to answering linear queries that combines the multiplicative weights update rule for no-regret learning and the exponential mechanism (McSherry & Talwar, 2007) for selecting queries. It is a simplification of the private multiplicative weights algorithm (Hardt & Rothblum, 2010). MWEM maintains a distribution over the data domain $\mathcal{X}$ and iteratively improves its approximation of the distribution given by the private dataset $\tilde{D}$. At each iteration, the algorithm privately selects a query q_t with approximately maximal error using the exponential mechanism and approximates the true answer to the query with Laplace noise (Dwork et al., 2006b). MWEM then improves the approximating distribution using the multiplicative weights update rule. This algorithm can be viewed as a two-player game in which a data player updates its distribution A_t using a no-regret online learning algorithm and a query player best responds using the exponential mechanism.

Our choice of extending MWEM stems from the following observations: (1) in the usual setting without public data, MWEM attains worst-case theoretical guarantees that are nearly information-theoretically optimal (Bun et al., 2018); (2) MWEM achieves state-of-the-art results in practice when it is computationally feasible to run; and (3) MWEM can be readily adapted to incorporate “prior” knowledge that is informed by public data.

However, maintaining a distribution A over a data domain $\mathcal{X} = \{0, 1\}^d$ is intractable when d is large, requiring a run-time of $O(n|\mathcal{Q}| + T|\mathcal{X}||\mathcal{Q}|))$, which is exponential in d (Hardt et al., 2012). Moreover, Ullman & Vadhan (2011) show that computational hardness is inherent for worst-case datasets, even in the case of 2-way marginal queries. Thus, applying MWEM is often impractical in real-world instances, prompting the development of new algorithms (Gaboardi et al., 2014; Vietri et al., 2020) that bypass computational barriers at the expense of some accuracy.

3.2. PMW$^{\mathsf{Pub}}$

We now introduce PMW$^{\mathsf{Pub}}$ in Algorithm 1, which adapts MWEM to utilize public data in the following ways:

First, the approximating distribution A_t is maintained over the public data domain $\widehat{\mathcal{X}}$ rather than $\mathcal{X}$, implying that the run-time of PMW$^{\mathsf{Pub}}$ is $O(n|\mathcal{Q}| + T|\widehat{\mathcal{X}}||\mathcal{Q}|))$. Because $|\widehat{\mathcal{X}}| \leq m$ is often significantly smaller than $|\mathcal{X}|$, PMW$^{\mathsf{Pub}}$ offers substantial improvements in both run-time and memory usage, scaling well to high-dimensional problems.

Second, A_0 is initialized to the distribution over $\widehat{\mathcal{X}}$ given by $\widehat{D}$. By default, MWEM initializes A_0 to be uniform over the data domain $\mathcal{X}$. This naïve prior is appropriate for worst-case analysis, but, in real-world settings, we can often form a reasonable prior that is closer to the desired distribution. Therefore, PMW$^{\mathsf{Pub}}$ initializes A_0 to match the distribution of $\widehat{D}$ under the assumption that the public dataset's distribution provides a better approximation of $\widetilde{D}$.

In addition, we make two additional improvements:

Permute-and-flip Mechanism. We replace the *exponential mechanism* with the *permute-and-flip mechanism* (McKenna & Sheldon, 2020), which like the *exponential mechanism*

Algorithm 1 PMW$^{\text{Pub}}$

```
Input: Private dataset $\widetilde{D} \in \mathcal{X}^n$, public dataset $\widehat{D} \in \mathcal{X}^m$, query class $\mathcal{Q}$, privacy parameter $\tilde{\varepsilon}$, number of iterations $T$.
Let the domain be $\widehat{\mathcal{X}} = \mathrm{supp}(\widehat{D})$.
Let size of the private dataset be $n = |\widetilde{D}|$.
Let $A_0$ be the distribution over $\widehat{\mathcal{X}}$ given by $\widehat{D}$
Initialize $\varepsilon_0 = \frac{\tilde{\varepsilon}}{\sqrt{2T}}$.
for $t = 1$ to $T$ do
    Sample query $q_t \in \mathcal{Q}$ using the permute-and-flip mechanism or exponential mechanism – i.e.,
        $\Pr[q_t] \propto \exp\left(\frac{\varepsilon_0 n}{2} |q(A_{t-1}) - q(\widetilde{D})|\right)$
    Measure: Let $a_t = q_t(\widetilde{D}) + \mathcal{N}\left(0, 1/n^2\varepsilon_0^2\right)$. (But, if $a_t < 0$, set $a_t = 0$; if $a_t > 1$, set $a_t = 1$.)
    Update: Let $A_t$ be a distribution over $\widehat{\mathcal{X}}$ s.t.
        $A_t(x) \propto A_{t-1}(x) \exp\left(q_t(x)\left(a_t - q_t(A_{t-1})\right)/2\right).$
end for
Output: $A = \mathrm{avg}_{t\in[T]} A_{t-1}$
```

runs in linear time but whose expected error is never higher.

Gaussian Mechanism. When taking measurements of sampled queries, we add Gaussian noise instead of Laplace noise. The Gaussian distribution has lighter tails, and in settings with a high degree of composition, the scale of Gaussian noise required to achieve some fixed privacy guarantee is lower (Canonne et al., 2020). Privacy guarantees for the *Gaussian mechanism* can be cleanly expressed in terms of concentrated differential privacy and the composition theorem given by Bun & Steinke (2016).

4. Theoretical Analysis

In this section, we analyze the accuracy of PMW$^{\text{Pub}}$ under the assumption that the public and private dataset are i.i.d. samples from two different distributions. The support of the a dataset $X \in \mathcal{X}^*$ is the set $\mathrm{supp}(X) = \{x \in \mathcal{X} : x \in X\}$, and we denote the support of the public dataset $\widehat{D}$ by $\widehat{\mathcal{X}} = \mathrm{supp}(\widehat{D})$. Recall that PMW$^{\text{Pub}}$ (Algorithm 1) takes as input a public dataset and then updates its distribution over the public dataset's support using the same procedure found in MWEM. We show that the accuracy of PMW$^{\text{Pub}}$ will depend on the best mixture error over the public dataset support $\widehat{\mathcal{X}}$, which we characterize using the best mixture error function $f_{\widetilde{D},\mathcal{Q}} : 2^{\mathcal{X}} \to [0,1]$ that measures a given support's ability to approximate the private dataset $\widetilde{D}$ over the set of queries $\mathcal{Q}$. The precise definition is as follows:

Definition 4.1. For any support $S \in 2^{\mathcal{X}}$, the best mixture error of S to approximate a dataset D over the queries Q is given by the function:

$$f_{D,Q}(S) = \min_{\mu\in\Delta(S)} \max_{q\in Q} \left| q\left(D\right) - \sum_{x\in S} \mu_x q(x) \right|$$

where $\mu \in \Delta(S)$ is a distribution over the set S with $\mu_x \geq 0$ for all $x \in S$ and $\sum_{x\in S} \mu_x = 1$.

Intuitively, PMW$^{\text{Pub}}$ reweights the public dataset in a differentially private manner to approximately match the private dataset's answers; the function $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$ captures how well the best possible reweighting on $\widehat{\mathcal{X}}$ would do in the absence of any privacy constraints. While running PMW$^{\text{Pub}}$ does not explicitly require calculating the best mixture error, in practice it may prove useful to release it in a privacy-preserving way. We present the following lemma, which shows that $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$ has bounded sensitivity.

Lemma 4.2. For any support $S \in 2^{\mathcal{X}}$ and set Q, the best mixture error function $f_{D,Q}$ is $\frac{1}{n}$ sensitive. That is for any pair of neighboring datasets D, D' of size n, $|f_{D,Q}(S) - f_{D',Q}(S)| \leq \frac{1}{n}$.

It follows that we can release $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$, using the Laplace or Gaussian mechanism with magnitude scaled by $\frac{1}{n}$.

We show that, if the public and private datasets are drawn from similar distributions, then, with high probability, $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$ is small. Note that the required size of the public dataset increases with the divergence between the private and public distributions.

Proposition 4.3. Let $\mu, \nu \in \Delta(\mathcal{X})$ be distributions with $\mathrm{D}_\infty(\mu\|\nu) < \infty$. Let $\widetilde{D} \sim \mu^n$ and $\widehat{D} \sim \nu^m$ be n and m independent samples from μ and ν respectively. Let $\widehat{\mathcal{X}}$ be the support of $\widehat{D}$. Let Q be a finite set of statistical queries $q : \mathcal{X} \to [0,1]$. Let $\alpha, \beta > 0$. If $n \geq \frac{8}{\alpha^2} \log\left(\frac{4|Q|}{\beta}\right)$ and $m \geq \left(\frac{32}{\alpha^2} e^{\mathrm{D}_2(\mu\|\nu)} + \frac{8}{3\alpha} e^{\mathrm{D}_\infty(\mu\|\nu)}\right) \log\left(\frac{4|Q|+4}{\beta}\right)$, then

$$\Pr\left[f_{\widetilde{D},Q}(\widehat{\mathcal{X}}) \leq \alpha\right] \geq 1 - \beta.$$

Proof. Note that we may assume $\alpha < 1$ as the result is trivial otherwise. Let $g(x) = \mu(x)/\nu(x)$. Then $0 \leq g(x) \leq e^{\mathrm{D}_\infty(\mu\|\nu)}$ for all x and, for $X \sim \nu$, we have $\mathbb{E}[g(X)] = 1$ and $\mathbb{E}[g(X)^2] = e^{\mathrm{D}_2(\mu\|\nu)}$. Define $\omega \in \Delta(\widehat{\mathcal{X}})$ by $\omega_x = \frac{g(x)}{\sum_{x\in\widehat{D}} g(x)}$ for $x \in \widehat{\mathcal{X}}$. Clearly $f_{\widetilde{D},Q}(\widehat{\mathcal{X}}) \leq \max_{q\in Q} \left| q(\widetilde{D}) - \sum_{x\in\widehat{D}} \omega_x q(x) \right|$.

Fix some $q \in Q$. By Hoeffding's inequality,

$$\Pr[|q(\widetilde{D}) - q(\mu)| \geq \alpha/4] \leq 2 \cdot e^{-\alpha^2 n/8}.$$

For $X \sim \nu$, $\mathbb{E}[g(X)q(X)] = q(\mu)$ and $\mathsf{Var}[g(X)q(X)] \leq \mathbb{E}[(g(X)q(X))^2] \leq \mathbb{E}[g(X)^2] = e^{\mathrm{D}_2(\mu\|\nu)}$. By Bernstein's inequality,

$$\Pr\left[\left|m \cdot q(\mu) - \sum_{x \in \widehat{D}} g(x)q(x)\right| \geq \frac{\alpha}{4}m\right]$$
$$\leq 2 \cdot \exp\left(\frac{-\alpha^2 m}{32 \cdot e^{\mathrm{D}_2(\mu\|\nu)} + \frac{8}{3}\alpha \cdot e^{\mathrm{D}_\infty(\mu\|\nu)}}\right).$$

Let $\hat{m} = \sum_{x \in \widehat{D}} g(x)$. Similarly,

$$\Pr\left[|\hat{m} - m| \geq \frac{\alpha}{4}m\right] = \Pr\left[\left|m - \sum_{x \in \widehat{D}} g(x)\right| \geq \frac{\alpha}{4}m\right]$$
$$\leq 2 \cdot \exp\left(\frac{-\alpha^2 m}{32 \cdot \left(e^{\mathrm{D}_2(\mu\|\nu)} - 1\right) + \frac{8}{3} \cdot \alpha \cdot e^{\mathrm{D}_\infty(\mu\|\nu)}}\right).$$

If all three of the events above do not happen, then

$$\left|q(\widetilde{D}) - \sum_{x \in \widehat{D}} \omega_x q(x)\right| = \left|\frac{1}{n}\sum_{x \in \widetilde{D}} q(x) - \frac{1}{\hat{m}}\sum_{x \in \widehat{D}} g(x)q(x)\right|$$
$$\leq \left|\frac{1}{n}\sum_{x \in \widetilde{D}} q(x) - q(\mu)\right| + \left|\frac{1}{\hat{m}}\left(mq(\mu) - \sum_{x \in \widehat{D}} g(x)q(x)\right)\right|$$
$$+ \frac{|\hat{m} - m|}{\hat{m}}|q(\mu)| \leq \frac{\alpha}{4} + \frac{\frac{\alpha}{4}m + \frac{\alpha}{4}m}{m - \frac{\alpha}{4}m} \leq \alpha.$$

Taking a union bound over all $q \in Q$ shows that the probability that any of these events happens is at most

$$2|Q| \cdot e^{-\alpha^2 n/8} + (2|Q| + 2) \cdot \exp\left(\frac{-\alpha^2 m}{32 \cdot e^{\mathrm{D}_2(\mu\|\nu)} + \frac{8}{3} \cdot \alpha \cdot e^{\mathrm{D}_\infty(\mu\|\nu)}}\right),$$

which is at most β if n and m are as large as the theorem requires. □

Having established sufficient conditions for good public data support, we bound the worst-case error of PMW$^{\mathsf{Pub}}$ running on a support $\widehat{\mathcal{X}}$. Since our method is equivalent to running MWEM on a restricted domain $\widehat{\mathcal{X}}$, its error bound will be similar to that of MWEM. Hardt et al. (2012) show that, if the number of iterations of the algorithm is chosen appropriately, then MWEM has error scaling with $\sqrt{\log(|\mathcal{X}|)}$ where $\mathcal{X}$ is the algorithm's data domain. Since PMW$^{\mathsf{Pub}}$ is initialized with the restricted data domain $\widehat{\mathcal{X}}$ based on a public dataset of size m, its error increases with $\sqrt{\log|\widehat{\mathcal{X}}|} \leq \sqrt{\log m}$ instead. Moreover, PMW$^{\mathsf{Pub}}$'s error bound includes the best-mixture error $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$. Taken together, we present the following bound:

Theorem 4.4. For any private dataset $\widetilde{D} \in \mathcal{X}^n$, set of statistical queries $Q \subset \{q : \mathcal{X} \to [0,1]\}$, public dataset $\widehat{D} \in \mathcal{X}^m$ with support $\widehat{\mathcal{X}}$, and privacy parameter $\tilde{\varepsilon} > 0$, PMW$^{\mathsf{Pub}}$ with parameter $T = \Theta\left(\frac{n\tilde{\varepsilon}\sqrt{\log m}}{\log|\mathcal{Q}|} + \log(1/\beta)\right)$ outputs a distribution A on $\widehat{\mathcal{X}}$ such that, with probability $\geq 1 - \beta$,

$$\max_{q \in \mathcal{Q}}\left|q(A) - q(\widetilde{D})\right|$$
$$\leq O\left(\sqrt{\frac{\log(|Q|) \cdot (\sqrt{\log m} + \log(\frac{1}{\beta}))}{n\tilde{\varepsilon}}} + f_{\widetilde{D},Q}\left(\widehat{\mathcal{X}}\right)\right).$$

4.1. Privacy Analysis

The privacy analysis follows from four facts: (i) Permute-and-flip satisfies ε_0-differential privacy (McKenna & Sheldon, 2020), which implies $\frac{1}{2}\varepsilon_0^2$-concentrated differential privacy. (ii) The Gaussian noise addition also satisfies $\frac{1}{2}\varepsilon_0^2$-concentrated differential privacy. (iii) The composition property of concentrated differential privacy allows us to add up these $2T$ terms (Bun & Steinke, 2016). (iv) Finally, we can convert the concentrated differential privacy guarantee into approximate differential privacy (Canonne et al., 2020).

Theorem 4.5. When run with privacy parameter $\tilde{\varepsilon} > 0$, PMW$^{\mathsf{Pub}}$ satisfies $\frac{1}{2}\tilde{\varepsilon}^2$-concentrated differential privacy and, for all $\delta > 0$, it satisfies$(\varepsilon(\delta), \delta)$-differential privacy, where

$$\varepsilon(\delta) = \inf_{\alpha > 1} \frac{1}{2}\tilde{\varepsilon}^2\alpha + \frac{\log(1/\alpha\delta)}{\alpha - 1} + \log(1 - 1/\alpha)$$
$$\leq \frac{1}{2}\tilde{\varepsilon}^2 + \sqrt{2\log(1/\delta)} \cdot \tilde{\varepsilon}.$$

5. Empirical Evaluation

In this section, we presents results comparing PMW$^{\mathsf{Pub}}$ against baseline algorithms[3] in a variety of settings using the American Census Survey and ADULT datasets.

5.1. Additional Baseline

DualQuery. Similar to MWEM, DualQuery (Gaboardi et al., 2014) frames query release as a two-player game, but it reverses the roles of the data and query players. Gaboardi et al. (2014) prove theoretical accuracy bounds for DualQuery that are worse than that of MWEM and show that on low-dimensional datasets where running MWEM is feasible, MWEM outperforms DualQuery. However, DualQuery employs optimization heuristics and is often more computationally efficient and scales to a wider range of query release problems than MWEM.

[3] HDMM has been considered as a relevant baseline algorithm in past query release work, but having consulted McKenna et al. (2018), we realized that running HDMM in many settings (including ours) is infeasible. We refer readers to Appendix A.5, where we provide a more detailed discussion.

5.2. Data

American Community Survey (ACS). We evaluate all algorithms on the 2018 American Community Survey (ACS), obtained from the IPUMS USA database (Ruggles et al., 2020). Collected every year by the US Census Bureau, the ACS provides statistics that capture the social and economic conditions of households across the country. Given that the Census Bureau may incorporate differential privacy into the ACS after 2025, the data provides a natural testbed for private query release algorithms in a real-world setting.

In total, we select 67 attributes,[4] giving us a data domain with dimension 287 and size $\approx 4.99 \times 10^{18}$. To run MWEM, we also construct a lower-dimensional version of the data. We refer to this data domain as ACS (reduced), which has dimension 33 and a size of 98304.

For our private dataset $\widetilde{D}$, we use the 2018 ACS for the state of Pennsylvania (PA-18) and Georgia (GA-18). To select our public dataset $\widehat{D}$, we explore the following:

Selecting across time. We consider the setting in which there exists a public dataset describing our population at a different point in time. Using the 2020 US Census release as an example, one could consider using the 2010 US Census as a public dataset for some differentially private mechanism. In our experiments, we use the ACS data for Pennsylvania and Georgia from 2010.

Selecting across states. We consider the setting in which there exists a public dataset collected concurrently from a different population. In the context of releasing state-level statistics, one can imagine for example that some states have differing privacy laws. In this case, we can identify data for a similar state that has been made public. In our experiments, we select a state with similar demographics to the private dataset's state—Ohio (OH-18) for Pennsylvania and North Carolina (NC-18) for Georgia. To explore how PMW$^{\text{Pub}}$ performs using public data from potentially more dissimilar distributions, we also run PMW$^{\text{Pub}}$ using the five largest states (by population) according to the 2010 US Census, i.e. California (CA-18), Texas (TX-18), New York (NY-18), Florida (FL-18), and Illinois (IL-18).

ADULT. We evaluate algorithms on the ADULT dataset from the UCI machine learning dataset repository (Dua & Graff, 2017). We construct private and public datasets by sampling with replacement rows from ADULT of size $0.9N$ and $0.1N$ respectively (where N is the number of rows in ADULT). Thus, we frame samples from ADULT as individuals from some population in which there exists both a public and private dataset trying to characterize it (with the former being significantly smaller). In total, the dataset has 13 attributes, and the data domain has dimension 146 and support size $\approx 7.32 \times 10^{11}$.

5.3. Empirical Optimizations

Following a remark made by Hardt et al. (2012) for optimizing the empirical performance of MWEM, we apply the multiplicative weights update rule using sampled queries q_i and measurements a_i from previous iterations i. However, rather than use all past measurements, we choose queries with estimated error above some threshold. Specifically at each iteration t, we calculate the term $c_i = |q_i(A_t) - a_i|$ for $i \leq t$. In random order, we apply multiplicative weights using all queries and measurements, indexed by i, where $c_i \geq \frac{c_t}{2}$, i.e. queries whose noisy error estimates are relatively high. In our implementation of MWEM and PMW$^{\text{Pub}}$, we use this optimization. We also substitute in the *permute-and-flip* and *Gaussian mechanisms* when running MWEM.

5.4. Hyperparameter tuning

On the ACS dataset, we select hyperparameters for PMW$^{\text{Pub}}$ using 5-run averages on the corresponding validation sets (treated as private) derived from the 2014 ACS release. Specifically, we evaluate Pennsylvania (PA-14) using PA-10 and OH-14, Georgia (GA-14) using GA-10 and NC-14, and both using CA-14, TX-14, NY-14, FL-14, and IL-14. In all other cases, we simply report the best performing five-run average across all hyperparameter choices. A list of hyperparameters is listed in Table 2 in the appendix.

5.5. Results

We first present results on the ACS data, demonstrating that PMW$^{\text{Pub}}$ achieves state-of-the-art performance in a real world setting in which there exist public datasets that come from slightly different distributions. Next, we run experiments on ADULT and vary how similar the public and private distributions are by artificially changing the proportion of females to males in the public dataset. Finally, we run additional experiments to highlight various aspects of PMW$^{\text{Pub}}$ in comparison to the baseline algorithms.

5.5.1. ACS

In Figure 1, we compare PMW$^{\text{Pub}}$ against baseline algorithms while using different public datasets. In addition, we plot the best mixture error function for each public dataset to approximate a lower bound on the error of PMW$^{\text{Pub}}$, which we estimate by running (non-private) multiplicative weights with early stopping (at 100 iterations).

We observe that on ACS (reduced) PA-18, MWEM achieves lower error than DualQuery at each privacy budget (Figure 1), supporting the view that MWEM should perform well when it is feasible to run it. Using PA-10, OH-18, and NY-18 as public datasets, PMW$^{\text{Pub}}$ improves upon the performance

[4] An inventory of attributes can be found in Appendix A.2.

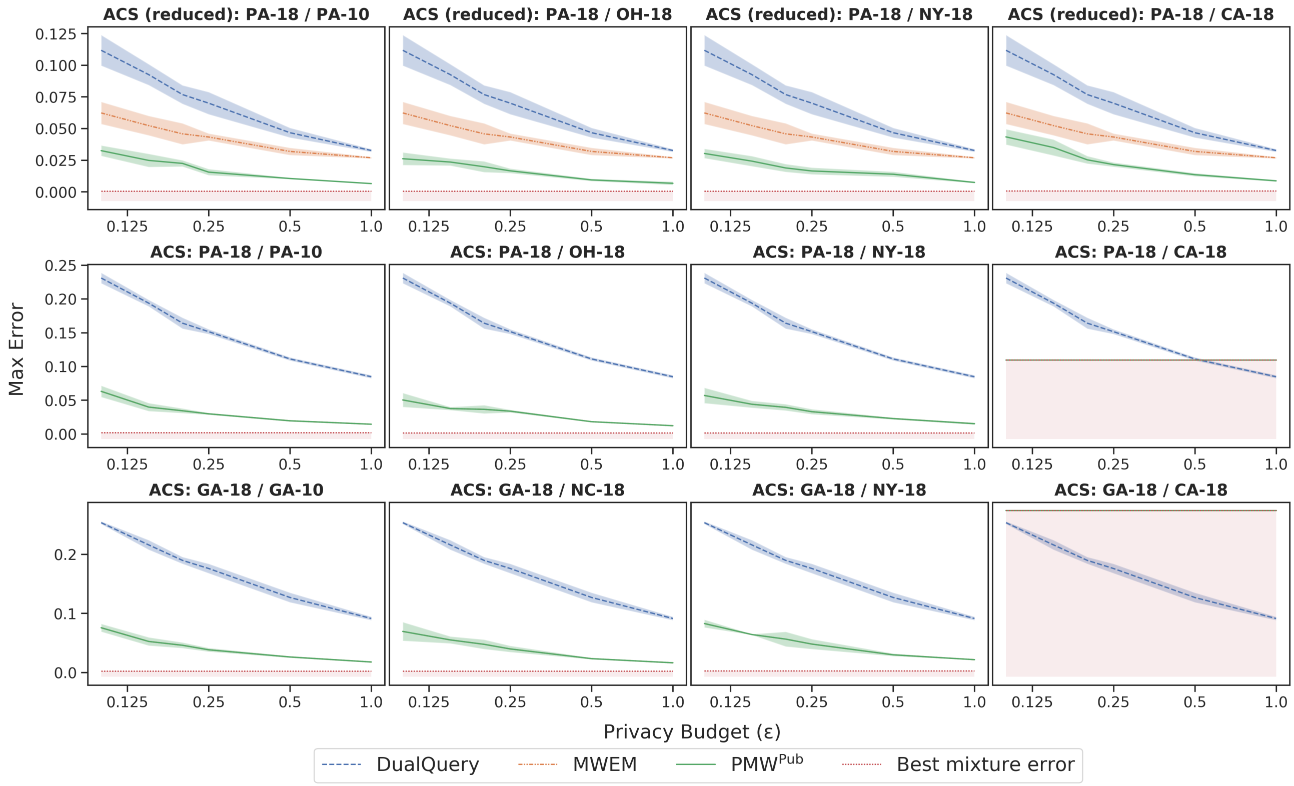

Figure 1: Max error for $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$. Results are averaged over 5 runs, and error bars represent one standard error. The *x-axis* uses a logarithmic scale. Given the support of each public dataset, we shade the area below the *best mixture error* to represent max error values that are unachievable by PMW$^{\text{Pub}}$. Additional results using our other choices of public datasets can found in Appendix A.4. **Top row:** 5-way marginals with a workload size of 3003 (maximum) on the 2018 ACS (reduced) for Pennsylvania. **Middle row:** 3-way marginals with a workload size of 4096 on the 2018 ACS for Pennsylvania. **Bottom row:** 3-way marginals with a workload size of 4096 on the 2018 ACS for Georgia.

of MWEM and outperforms all baselines. Similarly, on the full-sized ACS datasets for Pennsylvania and Georgia, PMW$^{\text{Pub}}$ outperforms DualQuery.

Next, we present results of PMW$^{\text{Pub}}$ when using CA-18 to provide examples where the distribution over the public dataset's support cannot be reweighted to answer all queries accurately. In Figure 1, we observe that when using CA-18, PMW$^{\text{Pub}}$ performs well on ACS (reduced) PA-18. However, on the set of queries defined for ACS PA-18 and GA-18, the best mixture error for CA-18 is high. Moreover, we observe that across all privacy budgets ε, PMW$^{\text{Pub}}$ achieves the best mixture error. Regardless of the number of rounds we run the algorithm for, the accuracy does not improve, and so the error plots in Figure 1 are flat and have no variance.

While it may be unsurprising that the support over a dataset describing California, a state with relatively unique demographics, is poor for answering large sets of queries on Pennsylvania and Georgia, one would still hope to identify this case ahead of time. One principled approach to verifying the quality of a public dataset is to *spend some privacy budget on measuring its best mixture error*. Given that finding the best mixture error is a sensitivity-$\frac{1}{n}$ query, we can use the *Laplace mechanism* to measure this value. For example, in the cases of both PA and GA (which have size $n \approx 10^5$), we can measure the best mixture error with a tiny fraction of the privacy budget (such as $\varepsilon = 0.01$) by adding Laplace noise with standard deviation $\frac{\sqrt{2}}{n\varepsilon} \approx 1.414 \times 10^{-3}$.

5.5.2. ADULT

To provide results on a different dataset, we also run experiments on ADULT in which we construct public and private datasets from the overall dataset. When sampled without bias, the public and private datasets come from the same distribution, and so the public dataset itself already approximates the distribution of the private dataset well. Consequently, we conduct additional experiments by sampling from ADULT according to the attribute *sex* with some bias. Specifically, we sample females with probability $r + \Delta$ where $r \approx 0.33$ is the proportion of females in the ADULT dataset. In Figure 2, we observe that running PMW$^{\text{Pub}}$ with

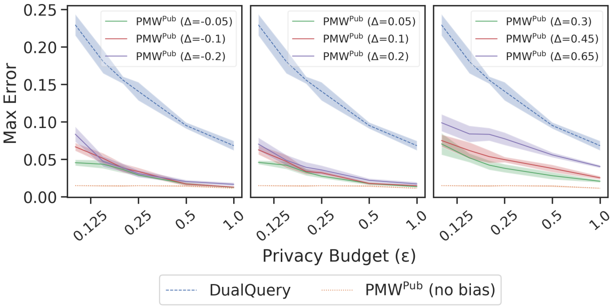

Figure 2: Max error on 3-way marginals across privacy budgets $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ where $\delta = \frac{1}{n^2}$ and the workload size is 256. Results are averaged over 5 runs, and error bars represent one standard error. Each public dataset is constructed by sampling from ADULT with some bias Δ over the attribute *sex* (labeled as PMW$^{\text{Pub}}$ (Δ)).

a public dataset sampled without bias ($\Delta = 0$) achieves very low error across all privacy budgets, and when using a public dataset sampled with low bias ($|\Delta| \leq 0.2$), PMW$^{\text{Pub}}$ still outperforms DualQuery. However, when the public dataset is extremely biased ($\Delta \in \{0.45, 0.65\}$), the performance of PMW$^{\text{Pub}}$ deteriorates (though it still significantly outperforms DualQuery). Therefore, we again show under settings in which the public and private distributions are relatively similar, PMW$^{\text{Pub}}$ achieves strong performance.

5.5.3. Additional empirical analysis

Public data size requirements. In Figure 3, we plot the performance on ACS PA-18 of PMW$^{\text{Pub}}$ against baseline solutions while varying the fraction of the public dataset used. Specifically, we sample some percentage ($p \in \{100\%, 10\%, 1\%, 0.1\%\}$) of rows from PA-10 and OH-18 to use as the public dataset. PMW$^{\text{Pub}}$ outperforms across all privacy budgets, even when only using 1% of the public dataset (Figure 3). From a practical standpoint, these results suggest that one can collect a public dataset that is relatively small (compared to the private dataset) and still achieve good performance using PMW$^{\text{Pub}}$.

Run-time. Although running MWEM on the ACS (reduced)-PA dataset is feasible, PMW$^{\text{Pub}}$ is computationally more efficient. An empirical evaluation can be found in Table 1.

6. Conclusion

In this paper, we study differentially private query release in which the privacy algorithm has access to both public and private data samples. We present our algorithm PMW$^{\text{Pub}}$, a variant of MWEM, that can take advantage of a source of public data. Unlike prior work however, we explore the case in which the public and private distributions are different, analyzing theoretical guarantees in this setting. Moreover, we demonstrate that PMW$^{\text{Pub}}$ improves accuracy over baseline algorithms in an empirical study involving the American Community Survey and ADULT datasets. In doing so, we also demonstrate that our algorithm is scalable to high-dimensional data.

For future work, one interesting avenue of research would be to extend other existing methods to *PAP* query release. However, we note that unlike MWEM, other methods such as DualQuery do not explicitly maintain a reweighting over a set of examples, which makes incorporating prior information less straightforward. In general, however, we reiterate that public data should be thought of as an additional source of prior information that lessens the burden of the private data. For example, one can imagine extending DualQuery by "pretraining" the approximating query distribution on

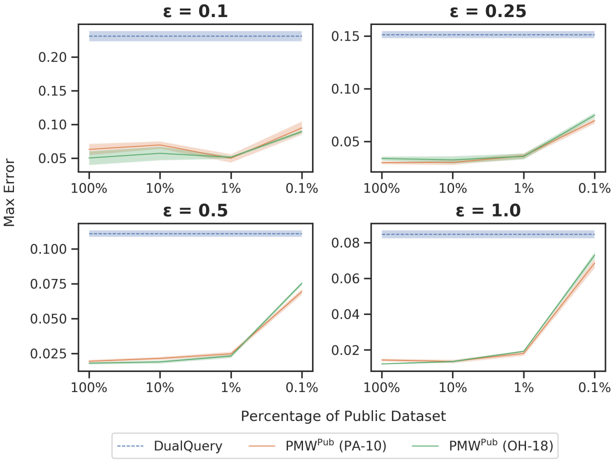

Figure 3: Performance comparison on ACS PA-18 while varying the size of the public dataset. We evaluate on 3-way marginals with a workload size of 4096 and privacy budgets defined by $\varepsilon \in \{0.1, 0.0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$.

Table 1: Run-time comparison between PMW$^{\text{Pub}}$ and MWEM on the 2018 ACS PA and ACS (reduced) PA, denoted as Full and Red. respectively. We compare the per-iteration run-time (in seconds) between PMW$^{\text{Pub}}$ (using PA-10 as the public dataset) and MWEM. Experiments are conducted using a single core on an i5-4690K CPU (3.50GHz) machine.

	Algo.	Per-iter. run-time
Red.	PMW$^{\text{Pub}}$	0.185
	MWEM	0.919
Full	PMW$^{\text{Pub}}$	2.021
	MWEM	–

public data first.[5] In addition, we contend that extending PMW$^{\text{Pub}}$ to other differential privacy problems, such as convex minimization problems (Ullman, 2015), is likewise interesting. In particular, incorporating public data into differentially private algorithms that generate synthetic data for supervised learning is an open research problem.

Acknowledgments

JU is supported by NSF grants CCF-1750640, CNS-1816028, CNS-1916020. ZSW is supported by NSF grant SCC-1952085 and a Google Faculty Research Award.

References

Abowd, J., Ashmead, R., Simson, G., Kifer, D., Leclerc, P., Machanavajjhala, A., and Sexton, W. Census topdown: Differentially private data, incremental schemas, and consistency with public knowledge. Technical report, Technical Report. US Census Bureau, 2019.

Abowd, J. M. The U.S. census bureau adopts differential privacy. In *ACM International Conference on Knowledge Discovery & Data Mining*, pp. 2867, 2018.

Alon, N., Bassily, R., and Moran, S. Limits of private learning with access to public data. *arXiv preprint arXiv:1910.11519*, 2019.

Bassily, R., Thakurta, A. G., and Thakkar, O. D. Model-agnostic private learning. *Advances in Neural Information Processing Systems*, 2018.

Bassily, R., Cheu, A., Moran, S., Nikolov, A., Ullman, J., and Wu, Z. S. Private query release assisted by public data. *arXiv preprint arXiv:2004.10941*, 2020a.

Bassily, R., Moran, S., and Nandi, A. Learning from mixtures of private and public populations. *arXiv preprint arXiv:2008.00331*, 2020b.

Beimel, A., Nissim, K., and Stemmer, U. Private learning and sanitization: Pure vs. approximate differential privacy. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, pp. 363–378. Springer, 2013.

Bun, M. and Steinke, T. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Proceedings of the 14th Conference on Theory of Cryptography*, TCC '16-B, pp. 635–658, Berlin, Heidelberg, 2016. Springer.

Bun, M., Ullman, J., and Vadhan, S. P. Fingerprinting codes and the price of approximate differential privacy. *SIAM J. Comput.*, 47(5):1888–1938, 2018. doi: 10.1137/15M1033587. URL https://doi.org/10.1137/15M1033587.

Canonne, C. L., Kamath, G., and Steinke, T. The discrete gaussian for differential privacy. In *NeurIPS*, 2020. URL https://arxiv.org/abs/2004.00010.

Dua, D. and Graff, C. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

Dwork, C. and Feldman, V. Privacy-preserving prediction. In *Conference On Learning Theory*, pp. 1693–1702. PMLR, 2018.

Dwork, C. and Rothblum, G. N. Concentrated differential privacy, 2016.

Dwork, C., Kenthapadi, K., McSherry, F., Mironov, I., and Naor, M. Our data, ourselves: Privacy via distributed noise generation. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pp. 486–503. Springer, 2006a.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. Calibrating noise to sensitivity in private data analysis. In *Proceedings of the 3rd Conference on Theory of Cryptography*, TCC '06, pp. 265–284, Berlin, Heidelberg, 2006b. Springer.

Gaboardi, M., Arias, E. J. G., Hsu, J., Roth, A., and Wu, Z. S. Dual query: Practical private query release for high dimensional data. In *International Conference on Machine Learning*, pp. 1170–1178, 2014.

Hardt, M. and Rothblum, G. N. A multiplicative weights mechanism for privacy-preserving data analysis. In *2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pp. 61–70. IEEE, 2010.

Hardt, M., Ligett, K., and McSherry, F. A simple and practical algorithm for differentially private data release. In *Advances in Neural Information Processing Systems*, pp. 2339–2347, 2012.

Ji, Z. and Elkan, C. Differential privacy based on importance weighting. *Machine learning*, 93(1):163–183, 2013.

McKenna, R. and Sheldon, D. Permute-and-flip: A new mechanism for differentially private selection, 2020.

[5] Gaboardi et al. (2014) show that, when compared to MWEM, DualQuery performs worse empirically and has worse theoretical accuracy bounds; the benefit of DualQuery over MWEM is computational efficiency (our variant PMW$^{\text{Pub}}$ is also efficient because the public data restricts the domain). We also observe that on ACS (reduced), MWEM outperforms DualQuery. Therefore, we believe that adapting MWEM to utilize public data is a more sensible endeavor.

McKenna, R., Miklau, G., Hay, M., and Machanavajjhala, A. Optimizing error of high-dimensional statistical queries under differential privacy. *PVLDB*, 11(10):1206–1219, 2018.

McSherry, F. and Talwar, K. Mechanism design via differential privacy. In *48th Annual IEEE Symposium on Foundations of Computer Science (FOCS'07)*, pp. 94–103. IEEE, 2007.

Nandi, A. and Bassily, R. Privately answering classification queries in the agnostic pac model. In *Algorithmic Learning Theory*, pp. 687–703. PMLR, 2020.

Papernot, N., Song, S., Mironov, I., Raghunathan, A., Talwar, K., and Úlfar Erlingsson. Scalable private learning with pate, 2018.

Ruggles, S. et al. Ipums usa: Version 10.0, doi: 10.18128/d010. *V10. 0*, 2020.

Ullman, J. Private multiplicative weights beyond linear queries. In *Proceedings of the 34th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, PODS '15, pp. 303–312, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450327572. doi: 10.1145/2745754.2745755. URL https://doi.org/10.1145/2745754.2745755.

Ullman, J. and Vadhan, S. Pcps and the hardness of generating private synthetic data. In *Theory of Cryptography Conference*, pp. 400–416. Springer, 2011.

Vietri, G., Tian, G., Bun, M., Steinke, T., and Wu, Z. S. New oracle-efficient algorithms for private synthetic data release. *arXiv preprint arXiv:2007.05453*, 2020.