LEVERAGING PUBLIC DATA FOR PRACTICAL PRIVATE QUERY RELEASE

Terrance Liu
Carnegie Mellon University
terrancl@cs.cmu.edu

Giuseppe Vietri
University of Minnesota
vietr002@umn.edu

Thomas Steinke
Google Brain
web@thomas-steinke.net

Jonathan Ullman
Northeastern University
jullman@ccs.neu.edu

Zhiwei Steven Wu
Carnegie Mellon University
zstevenwu@cmu.edu

ABSTRACT

In many statistical problems, incorporating priors can significantly improve performance. However, using prior knowledge in differentially private query release has remained underexplored, despite such priors commonly being available in the form of public data, 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, showing that $\mathsf{PMW}^{\mathsf{Pub}}$ outperforms state-of-the-art methods. Furthermore, our method scales well to high-dimensional data domains, where running many existing methods would be computationally infeasible.

1 INTRODUCTION

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. In this work, we study *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 practice, generating differentially private synthetic datasets is challenging without an excessively large private dataset, and a promising avenue for improving these algorithms is to incorporate *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 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., 2020), it crucially assumes that the public and private data come from the same distribution, and does not give efficient algorithms.

Our Contributions In light of these observations, we initiate the study of using public data to improve private query release in the more realistic setting where the public data distribution that is *related but not identical* to that of the private data. We make the following contributions:

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

[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.

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 $\widehat{\mathcal{X}} = \text{supp}(\widehat{D})$, the size of the private dataset be $n = |\widetilde{D}|$, 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: $A = \text{avg}_{t\in[T]} A_{t-1}$

2. We analyze the theoretical privacy and accuracy guarantees of PMW$^{\text{Pub}}$.
3. We empirically evaluate PMW$^{\text{Pub}}$ 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.
4. We show PMW$^{\text{Pub}}$ is computationally efficient and therefore is practical for much larger problem sizes than MWEM.

2 Public Data Assisted 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. MWEM maintains a distribution over the data domain $\mathcal{X}$ and iteratively improves its approximation of the distribution A_t given by the private dataset $\tilde{D}$. Our choice of extending MWEM stems from the following observations: (1) 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 runtime 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.

2.1 PMW$^{\text{Pub}}$

We introduce PMW$^{\text{Pub}}$ (Algorithm 1), which adapts MWEM to utilize public data in the following:

First, the approximating distribution A_t is maintained over the public data domain $\widehat{\mathcal{X}}$ rather than $\mathcal{X}$, implying that the runtime of PMW$^{\text{Pub}}$ 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$^{\text{Pub}}$ offers improvements in both runtime and memory usage.

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, $\mathsf{PMW}^{\mathsf{Pub}}$ initializes A_0 to match the distribution of $\widehat{D}$ under the assumption 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* runs in linear time but whose expected error is never higher.

Gaussian Mechanism. When measuring 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 expressed in terms of concentrated differential privacy and the composition theorem given by Bun & Steinke (2016).

2.2 Theoretical analysis

In this section, we analyze the accuracy of $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\widehat{\mathcal{X}} = \mathrm{supp}(\widehat{D})$. We show that the accuracy of $\mathsf{PMW}^{\mathsf{Pub}}$ 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 2.1. 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 $\sum_{x \in S} \mu_x = 1$.

Hardt et al. (2012) show MWEM has error scaling with $\sqrt{\log(|\mathcal{X}|)}$ where $\mathcal{X}$ is the data domain. Since $\mathsf{PMW}^{\mathsf{Pub}}$ is initialized with the restricted public data domain $\widehat{\mathcal{X}}$ of size m, its error increases with $\sqrt{\log|\widehat{\mathcal{X}}|} \leq \sqrt{\log m}$. Moreover, $\mathsf{PMW}^{\mathsf{Pub}}$'s error bound includes the best-mixture error $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$. Taken together, we present the following bound:

Theorem 2.2. 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 $\tilde{\varepsilon} > 0$, $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\geq 1-\beta$,

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

3 Empirical Evaluation

We evaluate on the 2018 American Community Survey (ACS), obtained from the IPUMS USA database (Ruggles et al., 2020). To run MWEM, we construct a lower-dimensional version of the data. We refer to this dataset as ACS (reduced). For our private dataset, we use the 2018 ACS for Pennsylvania (PA-18) and Georgia (GA-18). To select our public dataset, 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

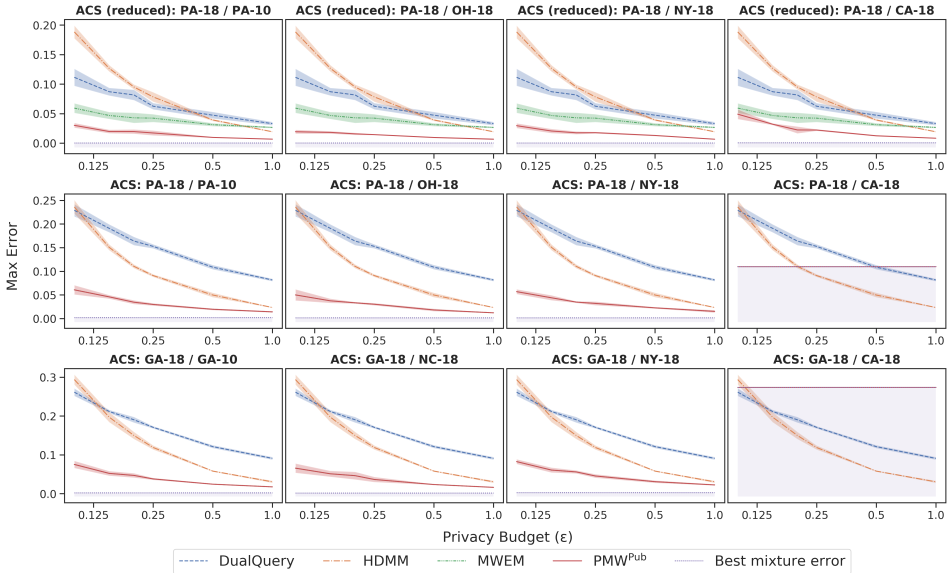

Figure 1: Max error for $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$. Results are averaged over 5 runs, and error bars represent one standard error. The *x-axis* uses a logarithmic scale. We shade the area below the *best mixture error* to represent max error values that are unachievable by $\mathsf{PMW}^{\mathsf{Pub}}$.

demographics to the private dataset's state—Ohio (OH-18) for Pennsylvania and North Carolina (NC-18) for Georgia. In addition, we select New York (NY-18) and California (CA-18).

3.1 Results

In Figure 1, we compare $\mathsf{PMW}^{\mathsf{Pub}}$ against baseline algorithm—DualQuery (Gaboardi et al., 2014) and HDMM (McKenna et al., 2018)—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 $\mathsf{PMW}^{\mathsf{Pub}}$, which we estimate by running (non-private) multiplicative weights with early stopping. On ACS (reduced), we evalutae on 5-way marginals with a workload size of 3003 (maximum). On the full-sized ACS, we evalute on 3-way marginals with a workload size of 4096.

We observe that on ACS (reduced) PA-18, MWEM achieves lower error than HDMM and DualQuery at $\varepsilon \leq 0.5$ (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, $\mathsf{PMW}^{\mathsf{Pub}}$ improves upon the performance of MWEM and outperforms all three baselines. Similarly, on the full-sized ACS datasets for Pennsylvania and Georgia, $\mathsf{PMW}^{\mathsf{Pub}}$ outperforms both baselines.

Next, we present results of $\mathsf{PMW}^{\mathsf{Pub}}$ 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, $\mathsf{PMW}^{\mathsf{Pub}}$ 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, $\mathsf{PMW}^{\mathsf{Pub}}$ achieves the best mixture error, regardless of the number of round we run the algorithm for.

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 (see Appendix A), we can use the *Laplace mechanism* to measure this value. For example, in the case 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 $\frac{\sqrt{2}}{n\varepsilon} \approx 1.414 \times 10^{-3}$.

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

John Abowd, Robert Ashmead, Garfinkel Simson, Daniel Kifer, Philip Leclerc, Ashwin Machanavajjhala, and William Sexton. Census topdown: Differentially private data, incremental schemas, and consistency with public knowledge. Technical report, Technical Report. US Census Bureau, 2019.

Raef Bassily, Kobbi Nissim, Adam Smith, Thomas Steinke, Uri Stemmer, and Jonathan Ullman. Algorithmic stability for adaptive data analysis. In *Proceedings of the forty-eighth annual ACM symposium on Theory of Computing*, pp. 1046–1059, 2016.

Raef Bassily, Albert Cheu, Shay Moran, Aleksandar Nikolov, Jonathan Ullman, and Zhiwei Steven Wu. Private query release assisted by public data. *arXiv preprint arXiv:2004.10941*, 2020.

Mark Bun and Thomas Steinke. 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.

Mark Bun, Jonathan Ullman, and Salil P. Vadhan. 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`.

Clément L. Canonne, Gautam Kamath, and Thomas Steinke. The discrete gaussian for differential privacy. In *NeurIPS*, 2020. URL `https://arxiv.org/abs/2004.00010`.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Cynthia Dwork and Guy N. Rothblum. Concentrated differential privacy, 2016.

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. 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.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. 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.

Marco Gaboardi, Emilio Jesús Gallego Arias, Justin Hsu, Aaron Roth, and Zhiwei Steven Wu. Dual query: Practical private query release for high dimensional data. In *International Conference on Machine Learning*, pp. 1170–1178, 2014.

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

Daniel Kifer. Consistency with external knowledge: The topdown algorithm, 2019. `http://www.cse.psu.edu/~duk17/papers/topdown.pdf`.

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

Ryan McKenna, Gerome Miklau, Michael Hay, and Ashwin Machanavajjhala. Optimizing error of high-dimensional statistical queries under differential privacy. *PVLDB*, 11(10):1206–1219, 2018.

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

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

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

Giuseppe Vietri, Grace Tian, Mark Bun, Thomas Steinke, and Zhiwei Steven Wu. New oracle-efficient algorithms for private synthetic data release. *arXiv preprint arXiv:2007.05453*, 2020.

A THEORY

A.1 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 A.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 $q_\phi(A) = \sum_{x\in\mathcal{X}} q(x)A(x)$.

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

Definition A.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 is a linear query specified by attributes $A = \{(A_i)_{i\in[k]} \mid A_1 \neq \ldots \neq A_k \in [d]\}$ and target $y \in (\mathcal{X}_1 \times \ldots \times \mathcal{X}_k)$, given by

$$\phi_{A,y}(x) = \begin{cases} 1 & : x_{a_1} = y_1 \wedge \ldots \wedge x_{a_k} = y_k \\ 0 & : \text{otherwise} \end{cases}$$

where $x_i \in \mathcal{X}_i$ means the ith attribute of record $x \in \mathcal{X}$. Each marginal has a total of $\prod_{i=1}^{k} |\mathcal{X}_{a_k}|$ queries, and we define a *workload* as a set of 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 A.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.

Definition A.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 A.5 (Concentrated DP)**.** A randomized mechanism $M : \mathcal{X}^n \to \mathcal{R}$ is $\frac{1}{2}\tilde{\epsilon}^2$-CDP, if for all neighboring datasets D, D' (i.e., differing on a single person), and for all $\alpha \in (1, \infty)$,

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

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 A.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 A.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

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

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

A.2 Theoretical Analysis

In this section, we analyze the accuracy of $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\mathsf{PMW}^{\mathsf{Pub}}$ (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 $\mathsf{PMW}^{\mathsf{Pub}}$ 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 A.7. 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, $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\mathsf{PMW}^{\mathsf{Pub}}$ 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 A.8. 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 A.9. 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,

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

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

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

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 $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\mathsf{PMW}^{\mathsf{Pub}}$ 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, $\mathsf{PMW}^{\mathsf{Pub}}$'s error bound includes the best-mixture error $f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$. Taken together, we present the following bound:

Theorem A.10. 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 $\tilde{\varepsilon} > 0$, $\mathsf{PMW}^{\mathsf{Pub}}$ 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 $\geq 1-\beta$,

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

A.3 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 A.11. When run with privacy parameter $\tilde{\varepsilon} > 0$, $\mathsf{PMW}^{\mathsf{Pub}}$ satisfies $\frac{1}{2}\tilde{\varepsilon}^2$-concentrated differential privacy and, for all $\delta > 0$, it satisfies$(\varepsilon(\delta), \delta)$-differential privacy, where

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

A.4 Additional proofs

Proposition A.12. Let $\tilde{D} \in \mathcal{X}^n$ and $\widehat{D} \in \mathcal{X}^m$. Let $\mathcal{Q}$ be a finite set of statistical queries $q : \mathcal{X} \to [0,1]$. Let $\tilde{\varepsilon} > 0$ and $T \in \mathbb{N}$. Let A be the output of Algorithm 1 with parameters $\tilde{\varepsilon}$ and T, query class $\mathcal{Q}$, and inputs $\tilde{D}$ as the private dataset and $\widehat{D}$ as the public dataset. Then A is a distribution on $\widehat{\mathcal{X}} = \text{supp}(\widehat{D}) \subset \mathcal{X}$. For all $\beta \in (0,1)$, if $T \geq 7\log(3/\beta)$, then

$$\Pr\left[\begin{array}{r}\max_{q\in\mathcal{Q}}|q(\widetilde{D}) - q(A)| \leq 2f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}}) + \sqrt{\frac{4\log m}{T} + \frac{4T}{\tilde{\varepsilon}^2 n^2} + \frac{4\sqrt{\log(3/\beta)}}{\tilde{\varepsilon}n}} \\ + \frac{2\sqrt{2T}}{\tilde{\varepsilon}n}\log|\mathcal{Q}| + \sqrt{\frac{1}{2T}\log\left(\frac{3}{\beta}\right)}\end{array}\right] \geq 1-\beta.$$

If we set $T = \Theta\left(\frac{\tilde{\varepsilon}n\sqrt{\log m}}{\log|\mathcal{Q}|} + \log(1/\beta)\right)$, then the bound above becomes

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

thus proving Theorem A.10.

Proof. We follow the analysis of Hardt et al. (2012). Let A_t, q_t, a_t be as in Algorithm 1. Let $\alpha_0 = f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}})$ be the error of the optimal reweighting of the public data. Let

$$D^* = \arg\min_{D\in\Delta(\widehat{\mathcal{X}})}\max_{q\in\mathcal{Q}}|q(D) - q(\widetilde{D})|$$

be the optimal reweighting so that $\max_{q\in\mathcal{Q}}|q(D^*) - q(\widetilde{D})| = \alpha_0$. We define a potential function $\Psi : \Delta(\widehat{\mathcal{X}}) \to \mathbb{R}$ by

$$\Psi(A) = \mathrm{D}_1(D^*\|A) = \sum_{x\in\widehat{\mathcal{X}}} D^*(x)\log\left(\frac{D^*(x)}{A(x)}\right).$$

Since Ψ is a KL divergence, it follows that, for all $A \in \Delta(\widehat{\mathcal{X}})$,

$$0 \leq \Psi(A) \leq \log\left(\frac{1}{\min_{x\in\widehat{\mathcal{X}}} A(x)}\right).$$

In particular, $\Psi(A_T) \geq 0$ and $\Psi(A_0) \leq \log m$, since any $x \in \widehat{\mathcal{X}}$ must be one of the m elements of $\widehat{D}$ and hence has $A_0(x) \geq 1/m$.

Fix an arbitrary $t \in [T]$. For all $x \in \widehat{\mathcal{X}}$, we have $A_t(x) = \frac{A_{t-1}(x)\exp(q_t(x)(a_t - q_t(A_{t-1}))/2)}{\sum_{y\in\widehat{\mathcal{X}}} A_{t-1}(y)\exp(q_t(y)(a_t - q_t(A_{t-1}))/2)}$. Thus

$$
\begin{aligned}
&\Psi(A_{t-1}) - \Psi(A_t) \\
&= \sum_{x\in\widehat{\mathcal{X}}} D^*(x) \log\left(\frac{A_t(x)}{A_{t-1}(x)}\right) \\
&= \sum_{x\in\widehat{\mathcal{X}}} D^*(x) \log\left(\frac{\exp(q_t(x)(a_t - q_t(A_{t-1}))/2)}{\sum_{y\in\widehat{\mathcal{X}}} A_{t-1}(y)\exp(q_t(y)(a_t - q_t(A_{t-1}))/2)}\right) \\
&= \sum_{x\in\widehat{\mathcal{X}}} D^*(x) q_t(x) \frac{a_t - q_t(A_{t-1})}{2} - \log\left(\sum_{y\in\widehat{\mathcal{X}}} A_{t-1}(y) \exp\left(q_t(y)\frac{a_t - q_t(A_{t-1})}{2}\right)\right) \\
&\geq q_t(D^*)\frac{a_t - q_t(A_{t-1})}{2} + 1 - \sum_{y\in\widehat{\mathcal{X}}} A_{t-1}(y)\exp\left(q_t(y)\frac{a_t - q_t(A_{t-1})}{2}\right) \\
&\hspace{20em} (\forall x > 0 \;\; \log x \leq x - 1) \\
&\geq q_t(D^*)\frac{a_t - q_t(A_{t-1})}{2} + 1 - \sum_{y\in\widehat{\mathcal{X}}} A_{t-1}(y)\left(1 + q_t(y)\frac{a_t - q_t(A_{t-1})}{2} + q_t(y)^2\frac{(a_t - q_t(A_{t-1}))^2}{4}\right) \\
&\hspace{20em} (\forall x \leq 1 \;\; \exp(x) \leq 1 + x + x^2) \\
&= q_t(D^*)\frac{a_t - q_t(A_{t-1})}{2} + 1 - 1 - q_t(A_{t-1})\frac{a_t - q_t(A_{t-1})}{2} - \mathop{\mathbb{E}}_{X\leftarrow A_{t-1}}\left[q_t(X)^2\right]\frac{(a_t - q_t(A_{t-1}))^2}{4} \\
&= (q_t(D^*) - q_t(A_{t-1}))\frac{a_t - q_t(A_{t-1})}{2} - \mathop{\mathbb{E}}_{X\leftarrow A_{t-1}}\left[q_t(X)^2\right]\frac{(a_t - q_t(A_{t-1}))^2}{4} \\
&\geq (q_t(D^*) - q_t(A_{t-1}))\frac{a_t - q_t(A_{t-1})}{2} - \frac{(a_t - q_t(A_{t-1}))^2}{4} \\
&= \frac{1}{4}(2q_t(D^*) - a_t - q_t(A_{t-1}))(a_t - q_t(A_{t-1})) \\
&= \frac{1}{4}(q_t(\widetilde{D}) - q_t(A_{t-1}))^2 + \frac{1}{2}(q_t(D^*) - q_t(\widetilde{D}))(a_t - q_t(A_{t-1})) - \frac{1}{4}(a_t - q_t(\widetilde{D}))^2 \\
&= \frac{1}{4}(q_t(\widetilde{D}) - q_t(A_{t-1}))^2 + \frac{1}{2}(q_t(D^*) - q_t(\widetilde{D}))(q_t(\widetilde{D}) - q_t(A_{t-1})) \\
&\qquad + \frac{1}{2}(q_t(D^*) - q_t(\widetilde{D}))(a_t - q_t(\widetilde{D})) - \frac{1}{4}(a_t - q_t(\widetilde{D}))^2 \\
&\geq \frac{1}{4}(q_t(\widetilde{D}) - q_t(A_{t-1}))^2 - \frac{1}{2}\alpha_0|q_t(\widetilde{D}) - q_t(A_{t-1})| \\
&\qquad + \frac{1}{2}(q_t(D^*) - q_t(\widetilde{D}))(a_t - q_t(\widetilde{D})) - \frac{1}{4}(a_t - q_t(\widetilde{D}))^2,
\end{aligned}
$$

where the final inequality follows from the fact that $|q_t(D^*) - q_t(\widetilde{D})| \leq \alpha_0$ by the definition of D^*.

Putting together what we have so far gives

$$\begin{aligned}\frac{2}{T}\log m &\geq \frac{2}{T}\left(\Psi(A_0)-\Psi(A_T)\right)\\ &= \frac{2}{T}\sum_{t\in[T]}\Psi(A_{t-1})-\Psi(A_t)\\ &\geq \frac{2}{T}\sum_{t\in[T]}\frac{1}{4}(q_t(\widetilde{D})-q_t(A_{t-1}))^2-\frac{2}{T}\sum_{t\in[T]}\frac{1}{2}\alpha_0|q_t(\widetilde{D})-q_t(A_{t-1})|\\ &\qquad+\frac{2}{T}\sum_{t\in[T]}\frac{1}{2}(q_t(D^*)-q_t(\widetilde{D}))(a_t-q_t(\widetilde{D}))-\frac{2}{T}\sum_{t\in[T]}\frac{1}{4}(a_t-q_t(\widetilde{D}))^2\\ &\geq \frac{1}{2}\left(\frac{1}{T}\sum_{t\in[T]}|q_t(\widetilde{D})-q_t(A_{t-1})|\right)^2-\frac{\alpha_0}{T}\sum_{t\in[T]}|q_t(\widetilde{D})-q_t(A_{t-1})|\\ &\qquad+\frac{1}{T}\sum_{t\in[T]}(q_t(D^*)-q_t(\widetilde{D}))(a_t-q_t(\widetilde{D}))-\frac{1}{2T}\sum_{t\in[T]}(a_t-q_t(\widetilde{D}))^2,\end{aligned}$$

where the final inequality uses the relationship between the 1-norm and 2-norm.

Now, for each $t\in[T]$ independently, $a_t-q_t(\widetilde{D})$ is distributed according to $\mathcal{N}(0,1/\varepsilon_0^2n^2)$. Thus the sum $\sum_{t\in[T]}(a_t-q_t(\widetilde{D}))^2$ follows a chi-square distribution with T degrees of freedom and mean $\frac{T}{\varepsilon_0^2n^2}$. This yields the tail bound

$$\forall\kappa\geq 1\quad \Pr\left[\sum_{t\in[T]}(a_t-q_t(\widetilde{D}))^2\geq\kappa\cdot\frac{T}{\varepsilon_0^2n^2}\right]\leq\left(\kappa\cdot e^{1-\kappa}\right)^{T/2}.$$

In addition, the noise $a_t-q_t(\widetilde{D})$ is independent from $q_t(D^*)-q_t(\widetilde{D})$. Hence, the sum $\sum_{t\in[T]}(q_t(D^*)-q_t(\widetilde{D}))(a_t-q_t(\widetilde{D}))$ follows a σ^2-subgaussian distribution with $\sigma^2=\frac{1}{\varepsilon_0^2n^2}\sum_{t\in[T]}(q_t(D^*)-q_t(\widetilde{D}))^2\leq\frac{T\alpha_0^2}{\varepsilon_0^2n^2}$. In particular,

$$\forall\lambda\geq 0\quad \Pr\left[\sum_{t\in[T]}(q_t(D^*)-q_t(\widetilde{D}))(a_t-q_t(\widetilde{D}))\geq\lambda\frac{\alpha_0\sqrt{T}}{\varepsilon_0 n}\right]\leq e^{-\lambda^2/2}.$$

Set $V:=\frac{1}{T}\sum_{t\in[T]}|q_t(\widetilde{D})-q_t(A_{t-1})|$.

Thus, for all $\kappa,\lambda\geq 0$,

$$\Pr\left[\frac{1}{2}V^2-\alpha_0V\leq\frac{2\log m}{T}+\frac{\kappa}{2\varepsilon_0^2n^2}+\frac{\lambda\alpha_0}{\varepsilon_0n\sqrt{T}}\right]\geq 1-\left(\kappa\cdot e^{1-\kappa}\right)^{T/2}-e^{-\lambda^2/2}.$$

The above expression contains the quadratic inequality $\frac{1}{2}V^2-\alpha_0V\leq\frac{2\log m}{T}+\frac{\kappa}{2\varepsilon_0^2n^2}+\frac{\lambda\alpha_0}{\varepsilon_0n\sqrt{T}}$. This equation implies

$$V\leq\alpha_0+\sqrt{\alpha_0^2+\frac{4\log m}{T}+\frac{\kappa}{\varepsilon_0^2n^2}+\frac{2\lambda\alpha_0}{\varepsilon_0n\sqrt{T}}}\leq 2\alpha_0+\sqrt{\frac{4\log m}{T}+\frac{\kappa}{\varepsilon_0^2n^2}+\frac{2\lambda\alpha_0}{\varepsilon_0n\sqrt{T}}}.$$

Now we invoke the properties of the permute-and-flip or exponential mechanism that selects q_t. For each $t\in[T]$, we have (Bassily et al., 2016, Lemma 7.1)

$$\mathop{\mathbb{E}}_{q_t}\left[|q_t(\widetilde{D})-q_t(A_{t-1})|\right]\geq\max_{q\in\mathcal{Q}}|q(\widetilde{D})-q(A_{t-1})|-\frac{2}{\varepsilon_0 n}\log|\mathcal{Q}|.$$

Since $0 \le |q_t(\widetilde{D}) - q_t(A_{t-1})| \le 1$, we can apply Azuma's inequality to obtain

$$\Pr\left[\frac{1}{T}\sum_{t\in[T]} |q_t(\widetilde{D}) - q_t(A_{t-1})| \ge \frac{1}{T}\sum_{t\in[T]} \max_{q\in\mathcal{Q}} |q(\widetilde{D}) - q(A_{t-1})| - \frac{2}{\varepsilon_0 n}\log|\mathcal{Q}| - \nu\right] \ge 1 - e^{-2\nu^2 T}$$

for all $\nu \ge 0$. Finally, for $A = \frac{1}{T}\sum_{t\in[T]} A_{t-1}$, we have

$$\begin{aligned}
\max_{q\in\mathcal{Q}} |q(\widetilde{D}) - q(A)| &\le \frac{1}{T}\sum_{t\in[T]} \max_{q\in\mathcal{Q}} |q(\widetilde{D}) - q(A_{t-1})| \\
&\le \frac{1}{T}\sum_{t\in[T]} |q_t(\widetilde{D}) - q_t(A_{t-1})| + \frac{2}{\varepsilon_0 n}\log|\mathcal{Q}| + \nu \\
&\qquad\qquad \text{(with probability } \ge 1 - e^{-2\nu^2 T}) \\
&\le 2\alpha_0 + \sqrt{\frac{4\log m}{T} + \frac{\kappa}{\varepsilon_0^2 n^2} + \frac{2\lambda\alpha_0}{\varepsilon_0 n\sqrt{T}}} + \frac{2}{\varepsilon_0 n}\log|\mathcal{Q}| + \nu. \\
&\qquad\qquad \text{(with probability } \ge 1 - \left(\kappa\cdot e^{1-\kappa}\right)^{T/2} - e^{-\lambda^2/2})
\end{aligned}$$

Now we set $\nu = \sqrt{\frac{1}{2T}\log\left(\frac{3}{\beta}\right)}$, $\kappa = 2$, and $\lambda = \sqrt{2\log(3/\beta)}$ and apply a union bound. If $T \ge 7\log(3/\beta)$, then

$$\Pr\left[\begin{array}{r} \max_{q\in\mathcal{Q}} |q(\widetilde{D}) - q(A)| \le 2\alpha_0 + \sqrt{\frac{4\log m}{T} + \frac{2}{\varepsilon_0^2 n^2} + \frac{2\sqrt{2\log(3/\beta)}\alpha_0}{\varepsilon_0 n\sqrt{T}}} \\ + \frac{2}{\varepsilon_0 n}\log|\mathcal{Q}| + \sqrt{\frac{1}{2T}\log\left(\frac{3}{\beta}\right)} \end{array}\right] \ge 1 - \beta.$$

Substituting in $\alpha_0 = f_{\widetilde{D},\mathcal{Q}}(\widehat{\mathcal{X}}) \le 1$ and $\varepsilon_0 = \frac{\tilde{\varepsilon}}{\sqrt{2T}}$ yields the result. ☐

We remark that the proof above uses the bound $\Psi(A_0) = \mathrm{D}_1\left(D^* \middle\| \widehat{D}\right) \le \log m$. This is tight in the worst case, but is likely to be loose in practice, as the private and public datasets are likely to be relatively similar. We could also alter Algorithm 1 to initialize A_0 to be uniform on $\widehat{\mathcal{X}}$, in which case we can replace $\log m$ with $\log|\widehat{\mathcal{X}}|$ in the final bound.

Lemma A.13. For any support $S \in 2^{\mathcal{X}}$ and set of linear queries 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, $|f_{D,Q}(S) - f_{D',Q}(S)| \le \frac{1}{n}$.

Proof. First, we show that the maximum of s-sensitive functions is an s-sensitive function and by symmetry the minimum of s-sensitive functions is s-sensitive. For any $s \le 1$, let $G = \{g : \mathcal{X} \to [0,1]\}$ be a class of s-sensitive functions and define a function $f : \mathcal{X} \to [0,1]$ as $f(X) = \max_{g\in G} g(X)$, for $X \in \mathcal{X}$.

Fix any support $S \in 2^{\mathcal{X}}$ and neighboring dataset D, D' with size n. Also fix the set Q and note each query $q \in Q$ is bounded in [0,1] and it's $\frac{1}{n}$-sensitive. Let $g' = \arg\max_{g\in G} g(D')$ and $g = \arg\max_{g\in G} g(D)$, then for neighboring D, D' we have

$$\begin{aligned}
f(D) - f(D') &\le f(D) - g(D') && \text{Since } f(D') \ge g(D') \\
&\le f(D) - g(D) + s && \text{Since } |g(D) - g(D')| \le s \\
&= s && \text{Since } f(D) = g(D)
\end{aligned}$$

Similarly, we can show that $f(D') - f(D) \le s$, therefore f is s-sensitive.

Since a marginal query $q \in \mathcal{Q}$, is $\frac{1}{n}$-sensitive, after fixing any μ the expression

$$\max_{q\in Q}\left|q(D) - \sum_{x\in S}\mu_x q(x)\right|$$

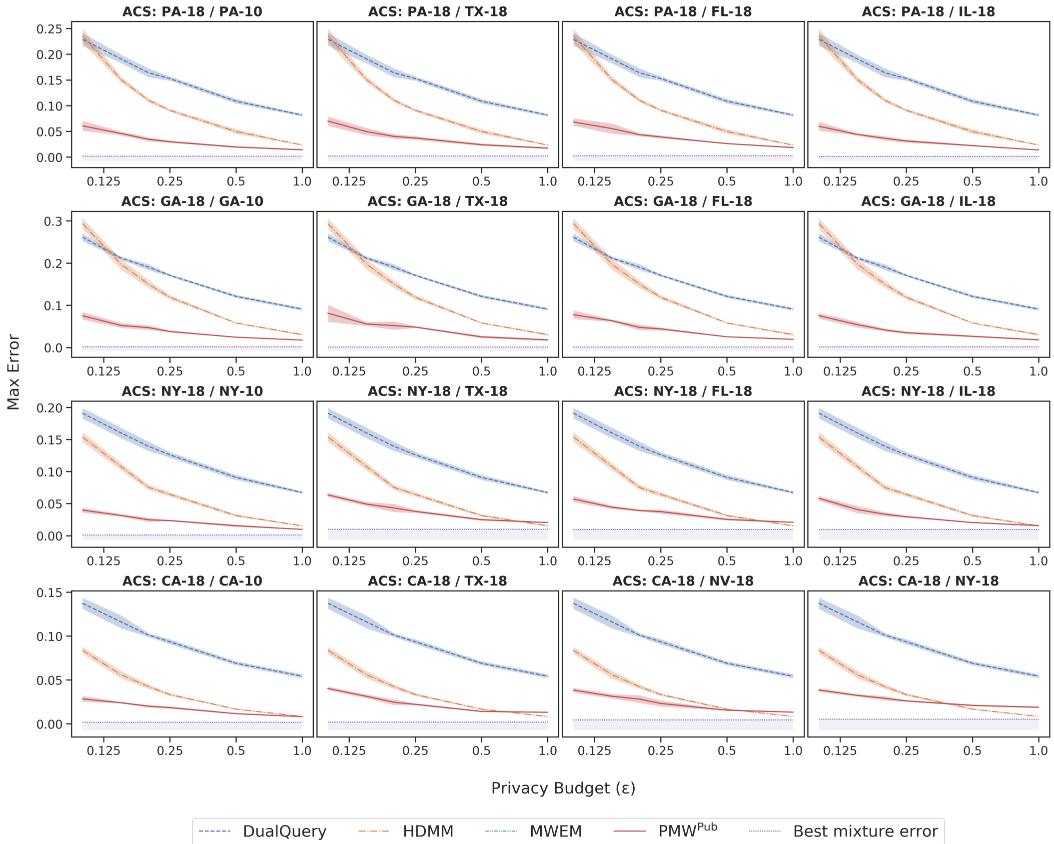

Figure 2: Additional plots of the max error (3-way marginals and workload size of 4096) for $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$ on PA-18 (Row 1), GA-18 (Row 2), NY-18 (Row 3), and CA-18 (Row 4). Results are averaged over 5 runs, and error bars represent one 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 $\mathsf{PMW}^{\mathsf{Pub}}$.

is a max of $\frac{1}{n}$ sensitive functions, then by the argument above it is a $\frac{1}{n}$-sensitive function. It follows that $f_{D,Q}(S)$ is a minimum of $\frac{1}{n}$-sensitive functions therefore $f_{D,Q}(S)$ is $\frac{1}{n}$-sensitive.

□

B Additional empirical evaluations

B.1 Empirical evaluation on ACS

We first provide additional experiments run on the 2018 ACS dataset.

B.1.1 Evaluation on additional data for states

In Figure 2, we plot results for ACS PA-18 and ACS GA-18 comparing $\mathsf{PMW}^{\mathsf{Pub}}$ using the 2010 ACS data (PA-10 and GA-10) with the public datasets Texas (TX-18), Florida (FL-18(, and Illinois (IL-18). Together with California and New York, these three states make up the five largest states by population according to the 2010 U.S. Census. In addition, we present results on 2018 ACS data for the states of New York (NY-18) and California (CA-18). To run $\mathsf{PMW}^{\mathsf{Pub}}$, we choose Texas (TX-18), Florida (FL-18), and Illinois (IL-18) for New York and choose Texas (TX-18), Nevada (NV-18), and New York (NY-18) for California.

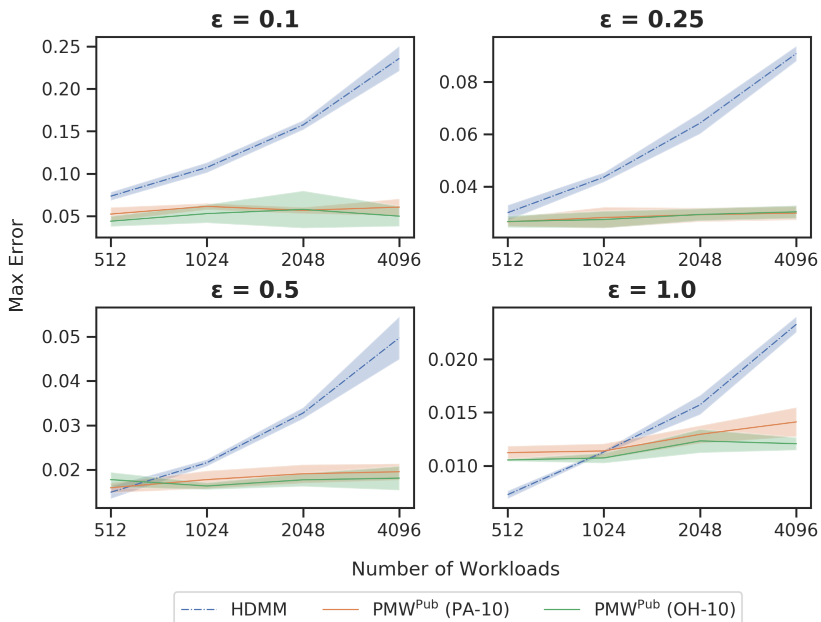

Figure 3: Performance comparison of $\mathsf{PMW}^{\mathsf{Pub}}$ against HDMM on 3-way marginals on ACS PA-18 while varying the number of workloads. We evaluate on privacy budgets $\varepsilon \in \{0.1, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$ and present results of $\mathsf{PMW}^{\mathsf{Pub}}$ using ACS PA-10 and OH-18.

B.1.2 Additional analysis of $\mathsf{PMW}^{\mathsf{Pub}}$

Workload scalability. On the 2018 ACS dataset for Pennsylvania, HDMM scales poorly with respect to workload size when compared to $\mathsf{PMW}^{\mathsf{Pub}}$. Figure 3 shows that although the maximum error of HDMM grows significantly as we increase the number of 3-way marginal queries, the maximum error of $\mathsf{PMW}^{\mathsf{Pub}}$ remains relatively stable. Our experiments suggest that in settings in which the goal is to release very large workloads of queries, $\mathsf{PMW}^{\mathsf{Pub}}$ may be a more suitable algorithm for achieving high accuracy.

Public data size requirements. In Figure 4, we plot the performance on ACS PA-18 of $\mathsf{PMW}^{\mathsf{Pub}}$ 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. $\mathsf{PMW}^{\mathsf{Pub}}$ outperforms both baselines across all privacy budgets, even when only using 1% of the public dataset (Figure 4). 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 $\mathsf{PMW}^{\mathsf{Pub}}$.

Run-time. Although running MWEM on the ACS (reduced)-PA dataset is feasible, $\mathsf{PMW}^{\mathsf{Pub}}$ is computationally more efficient. However, as a non-iterative algorithm, HDMM runs significantly faster, presenting a trade-off between the run-time and performance of the two algorithms. An empirical evaluation can be found in Table 1 in the appendix.

B.1.3 Run-time

To numerically compare the computational efficiency of $\mathsf{PMW}^{\mathsf{Pub}}$ vs. HDMM, we present run-times on the ACS (reduced)-PA dataset in Table 1.

B.1.4 Using the last iterate

In this work, we present theoretical guarantees of $\mathsf{PMW}^{\mathsf{Pub}}$ in which we output the average distribution $A = \text{avg}_{t \leq T} A_t$ (see Algorithm 1), mimicking the output in the original formulation of MWEM. However, Hardt et al. (2012) note that while they prove guarantees for this variant of MWEM, in

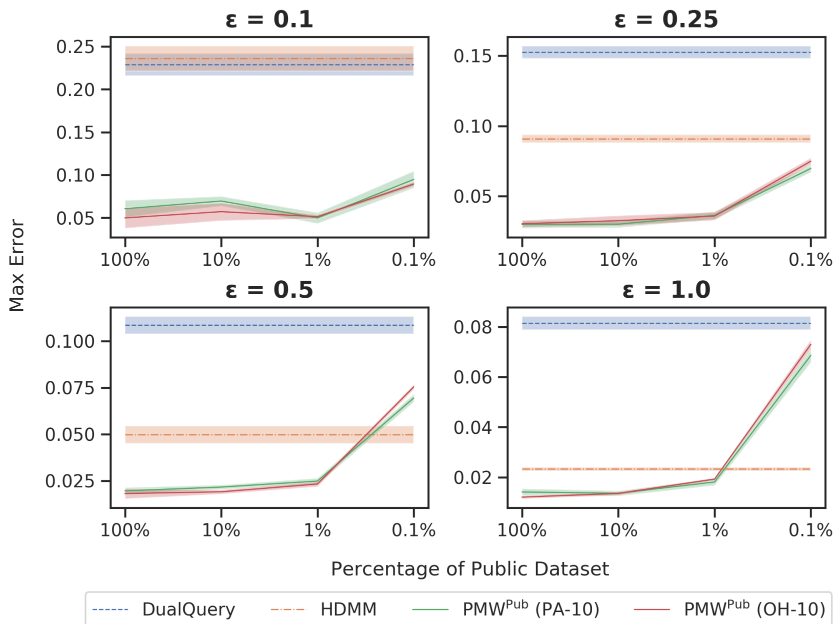

Figure 4: Performance comparison on ACS PA-18 while varying the size of the public dataset. We evaluate on 3-way marginals with a workload size of 4096 and privacy budgets defined by $\varepsilon \in \{0.1, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$.

Table 1: Run-time comparison between $\textsf{PMW}^{\textsf{Pub}}$, MWEM, and HDMM on the 2018 ACS PA and ACS (reduced) PA, denoted as FULL and Red. respectively. We compare the per-iteration run-time time between $\textsf{PMW}^{\textsf{Pub}}$ (using PA-10 as the public dataset) and MWEM, as well as the total run-time across all three algorithms. We express the total run-time of $\textsf{PMW}^{\textsf{Pub}}$ and MWEM in terms of the hyperparameter T, which determines the number of iterations we run each algorithm for. Experiments are conducted using a single core on a i5-4690K CPU (3.50GHz) machine.

		TIME (SECONDS)	
	ALGO.	PER-ITER.	TOTAL
RED.	$\textsf{PMW}^{\textsf{Pub}}$	0.185	$0.185 \times T$
	MWEM	0.919	$0.919 \times T$
	HDMM	–	23.841
FULL	$\textsf{PMW}^{\textsf{Pub}}$	2.021	$2.021 \times T$
	MWEM	–	–
	HDMM	–	24.236

practical settings, one can often achieve better results by outputting the distribution from the last iterate, A_T. In Figure 5, we compare $\textsf{PMW}^{\textsf{Pub}}$ to the variant of $\textsf{PMW}^{\textsf{Pub}}$ that outputs A_T and observe that indeed, outputting the last iterate achieves better performance across all experiments (excluding those in which the best mixture error of the public dataset's support is high, i.e. CA-18).

B.1.5 IDENTIFYING PUBLIC DATASETS WITH POOR SUPPORT

In Section 3.1, we describe how using Laplace noise, one can get determine the quality of a support by getting a noisy estimate of the best mixture error for any public dataset. While we emphasize that this strategy is the most principled approach to ensuring the public data is viable for $\textsf{PMW}^{\textsf{Pub}}$, we note that in settings like ours in which we have a validation set, one can apply additional sanity checks. For instance, in Figure 6, we observe that $\textsf{PMW}^{\textsf{Pub}}$ performs poorly on the validation set when using CA-14, both in absolute terms and relative to the other public datasets. For demonstra-

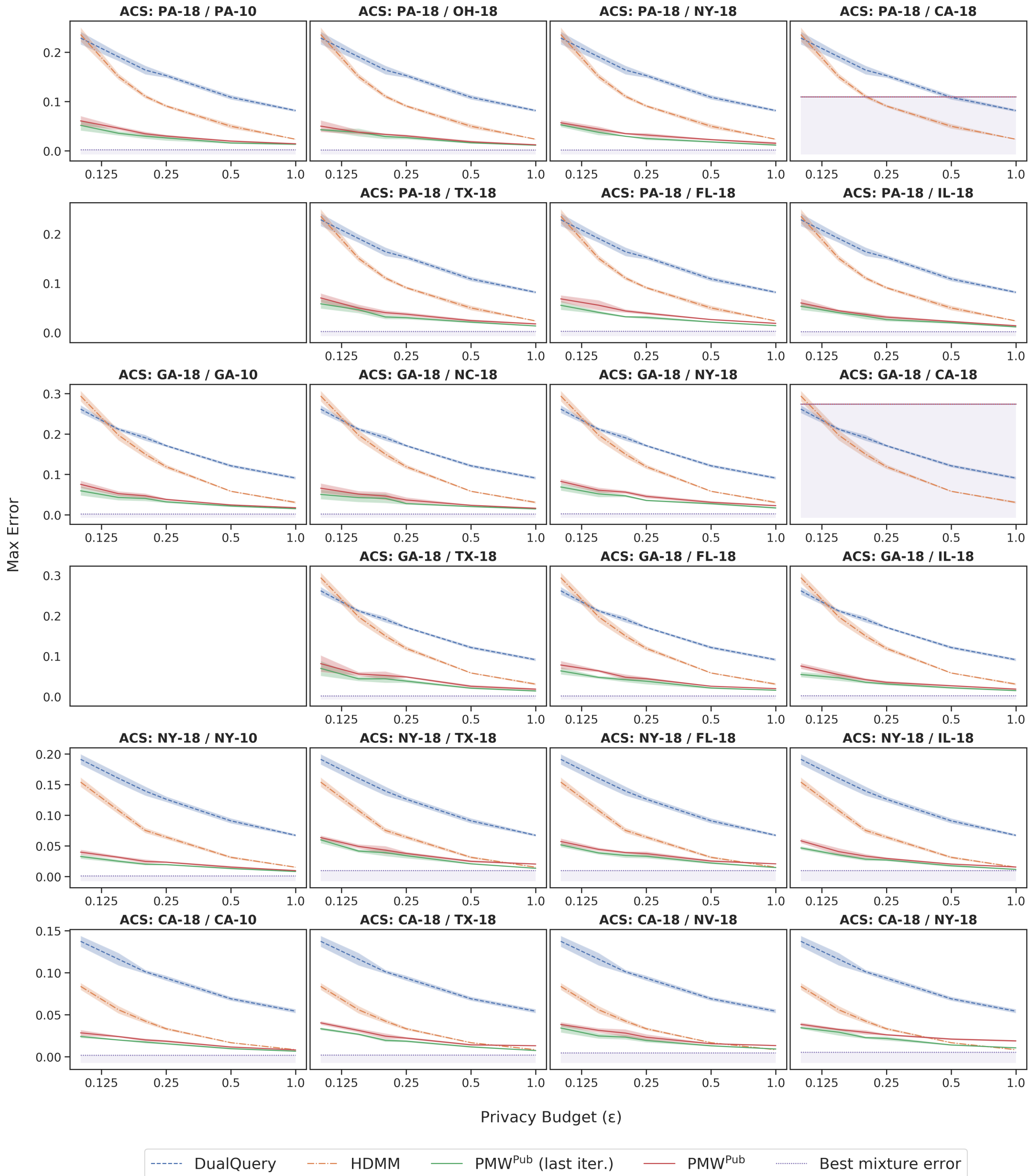

Figure 5: We compare $\mathsf{PMW}^{\mathsf{Pub}}$ with the variant of $\mathsf{PMW}^{\mathsf{Pub}}$ that outputs the last iterate A_T for all experiments (3-way marginals and workload size of 4096) on the (full-sized) 2018 ACS dataset, plotting max error for $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ 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 $\mathsf{PMW}^{\mathsf{Pub}}$. Using the last iterate in $\mathsf{PMW}^{\mathsf{Pub}}$ improves performance across all experiments.

tion purposes, we show in Table 2 that if we select the public dataset (at each privacy budget ε) based solely on which public dataset performed best on the validation set, we achieve very strong results in comparison to both baselines. Thus in practical settings, one can use validation sets in conjunction with the best mixture error function to find a suitable public dataset (for example, one can first filter

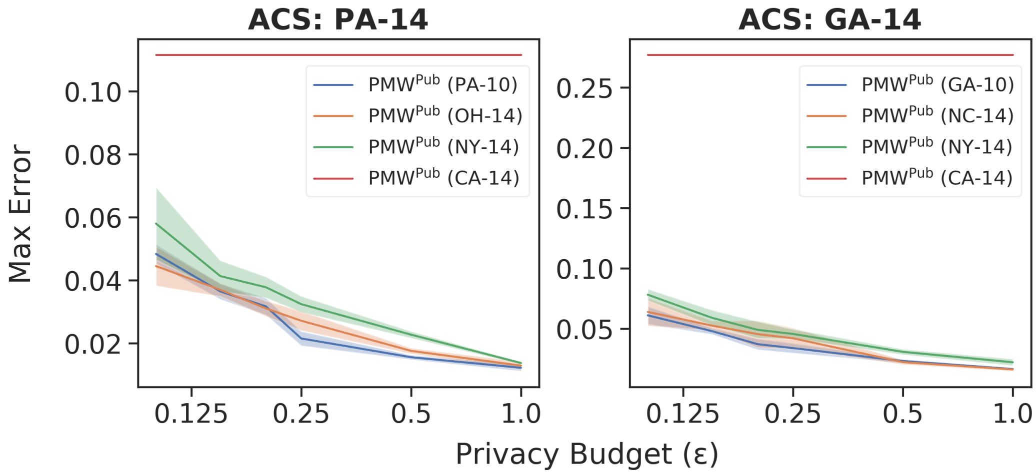

Figure 6: Max error on the ACS validation sets for 3-way marginals with a workload size of 4096 with privacy $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$. Results are averaged over 5 runs, and error bars represent one standard error. The *x-axis* uses a logarithmic scale. **Left:** 2014 ACS for Pennsylvania. **Right:** 2014 ACS for Georgia.

Table 2: Max error (averaged over 5 runs, best results in **bold**) comparison on the 2018 ACS (reduced)-PA, 2018 ACS-PA, and 2018 ACS-GA datasets. At each privacy budget parametrized by $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ and $\delta = \frac{1}{n^2}$, PMW^{Pub} uses the public dataset (and corresponding hyperparameter T) that achieves the lowest max error on the validation set.

Dataset	Algo.	$\varepsilon = 0.1$	$\varepsilon = 0.15$	$\varepsilon = 0.2$	$\varepsilon = 0.25$	$\varepsilon = 0.5$	$\varepsilon = 1$
ACS (red.)-PA	PMW^{Pub}	**0.0301**	**0.0197**	**0.0196**	**0.0172**	**0.0097**	**0.0067**
	HDMM	0.1883	0.1267	0.0951	0.0782	0.0392	0.0193
	DualQuery	0.1115	0.0871	0.0816	0.0625	0.0473	0.0330
ACS-PA	PMW^{Pub}	**0.0499**	**0.0458**	**0.0332**	**0.0298**	**0.0195**	**0.0141**
	HDMM	0.2360	0.1506	0.1104	0.0908	0.0497	0.0233
	DualQuery	0.2289	0.1908	0.1639	0.1526	0.1086	0.0816
ACS-GA	PMW^{Pub}	**0.0753**	**0.0523**	**0.0470**	**0.0380**	**0.0244**	**0.0175**
	HDMM	0.2939	0.1972	0.1500	0.1190	0.0581	0.0306
	DualQuery	0.2615	0.2117	0.1904	0.1709	0.1212	0.0910

out poor public datasets using a validation set and then find the best mixture error of any remaining candidates).

B.2 Empirical evaluation on 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.

B.2.1 Data

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).

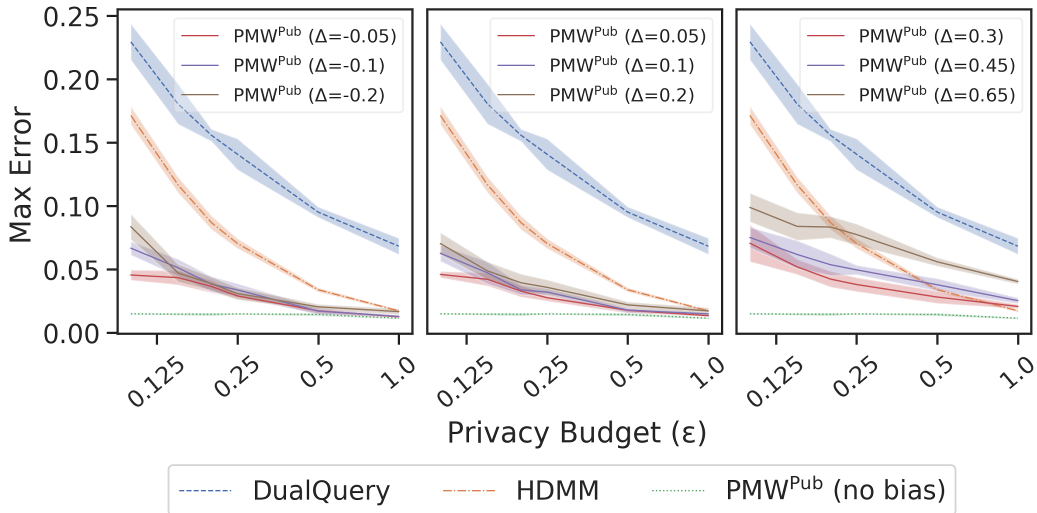

Figure 7: Max error on 3-way marginals across privacy budgets $\varepsilon \in \{0.1, 0.15, 0.2, 0.25, 0.5, 1\}$ where $\delta = \frac{1}{n^2}$ and the workload size is 256. Results are averaged over 5 runs, and error bars represent one standard error. Each public dataset is constructed by sampling from ADULT with some bias Δ over the attribute *sex* (labeled as $\mathsf{PMW}^{\mathsf{Pub}}$ (Δ)).

B.2.2 Results

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 7, we observe that running $\mathsf{PMW}^{\mathsf{Pub}}$ with 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 ($|\Delta| \leq 0.2$), $\mathsf{PMW}^{\mathsf{Pub}}$ still outperforms both baselines. However, when the public dataset is extremely biased ($\Delta \in \{0.45, 0.65\}$), the performance of $\mathsf{PMW}^{\mathsf{Pub}}$ deteriorates, with HDMM outperforming it at $\varepsilon \in \{0.5, 1\}$. Therefore, we again show under settings in which the public and private distributions are relatively similar, $\mathsf{PMW}^{\mathsf{Pub}}$ achieves strong performance.

C Experimental setup details

We provide additional information regarding our experiments

C.1 Dataset

ACS In total, we select 67 attributes, 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, we use the 2018 ACS for the state of Pennsylvania (PA-18) and Georgia (GA-18). To select our public dataset, we explore the following:

ADULT. In total, the dataset has 13 attributes, and the data domain has dimension 146 and support size $\approx 7.32 \times 10^{11}$.

Attributes for our experiments on ACS, ACS (reduced), and ADULT:

- **ACS:** ACREHOUS, AGE, AVAILBLE, CITIZEN, CLASSWKR, DIFFCARE, DIFFEYE, DIFFHEAR, DIFFMOB, DIFFPHYS, DIFFREM, DIFFSENS, DIVINYR, EDUC, EMPSTAT, FERTYR, FOODSTMP, GRADEATT, HCOVANY, HCOVPRIV, HINSCAID,

HINSCARE, HINSVA, HISPAN, LABFORCE, LOOKING, MARRINYR, MARRNO, MARST, METRO, MIGRATE1, MIGTYPE1, MORTGAGE, MULTGEN, NCHILD, NCHLT5, NCOUPLES, NFATHERS, NMOTHERS, NSIBS, OWNERSHP, RACAMIND, RACASIAN, RACBLK, RACE, RACOTHER, RACPACIS, RACWHT, RELATE, SCHLTYPE, SCHOOL, SEX, SPEAKENG, VACANCY, VEHICLES, VET01LTR, VET47X50, VET55X64, VET75X90, VET90X01, VETDISAB, VETKOREA, VETSTAT, VETVIETN, VETWWII, WIDINYR, WORKEDYR

- **ACS (reduced):** DIFFEYE, DIFFHEAR, EMPSTAT, FOODSTMP, HCOVPRIV, HINSCAID, HINSCARE, OWNERSHP, RACAMIND, RACASIAN, RACBLK, RACOTHER, RACPACIS, RACWHT, SEX
- **ADULT:** sex, income>50K, race, relationship, marital-status, workclass, occupation, education-num, native-country, capital-gain, capital-loss, hours-per-week, age

In addition, we discretize the following continuous attributes (with the number of bins after preprocessing) into categorical attributes:

- **ACS:** AGE (10)
- **ACS (reduced):** AGE (10)
- **ADULT:** capital-gain (16), capital-loss (6), hours-per-week (10), age (10)

C.2 Hyperparameters

On the ACS dataset, we select hyperparameters for $\mathsf{PMW}^{\mathsf{Pub}}$ 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. We report hyperparameters used across all experiments in Table 3. Note that HDMM does not have hyperparameters.

Table 3: Hyperparameter selection for experiments on all datasets.

Method	Parameter	Values
$\mathsf{PMW}^{\mathsf{Pub}}$	T	300, 250, 200, 150, 125, 100, 75, 50, 25, 10, 5
MWEM	T	300, 250, 200, 150, 125, 100, 75, 50, 25, 10, 5
DualQuery	samples η	500 250 100 50 5 4 3 2

C.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 $\mathsf{PMW}^{\mathsf{Pub}}$, we use this optimization. We also substitute in the *permute-and-flip* and *Gaussian mechanisms* when running MWEM.

C.4 Baselines

We provide additional background information regarding our baseline algorithms.

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.

HDMM. Unlike MWEM and DualQuery, which solve the query release problem by generating synthetic data, the High-Dimensional Matrix Mechanism (McKenna et al., 2018) is designed to directly answer a workload of queries. By representing query workloads compactly, HDMM selects a new set of "strategy" queries that minimize the estimated error with respect to the input workload. The algorithm then answers the "strategy" queries using the *Laplace mechanism* and reconstructs the answers to the input workload queries using these noisy measurements. With the US Census Bureau incorporating HDMM into its releases (Kifer, 2019), the algorithm offers a particularly suitable baseline for privately answering statistical queries on the ACS dataset.