PERSONALIZED DIFFERENTIAL PRIVACY FOR RIDGE REGRESSION

Anonymous authors
Paper under double-blind review

ABSTRACT

The increased application of machine learning (ML) in sensitive domains requires protecting the training data through privacy frameworks, such as differential privacy (DP). DP requires to specify a uniform privacy level ε that expresses the maximum privacy loss that each data point in the entire dataset is willing to tolerate. Yet, in practice, different data points often have different privacy requirements. Having to set one uniform privacy level is usually too restrictive, often forcing a learner to guarantee the stringent privacy requirement, at a large cost to accuracy. To overcome this limitation, we introduce our novel Personalized-DP Output Perturbation method (PDP-OP) that enables to train Ridge regression models with individual per data point privacy levels. We provide rigorous privacy proofs for our PDP-OP as well as accuracy guarantees for the resulting model. This work is the first to provide such theoretical accuracy guarantees when it comes to personalized DP in machine learning, whereas previous work only provided empirical evaluations. We empirically evaluate PDP-OP on synthetic and real datasets and with diverse privacy distributions. We show that by enabling each data point to specify their own privacy requirement, we can significantly improve the privacy-accuracy trade-offs in DP. We also show that PDP-OP outperforms the personalized privacy techniques of Jorgensen et al. (2015).

1 INTRODUCTION

Over the last decade, the amount of private data collected about individuals has experienced an exponential growth. As the data is used for computing statistics, training recommender systems, and automated decision-making in sensitive domains, such as medicine, privacy concerns around this data are growing. The gold standard for analyzing the data with privacy guarantees is Differential Privacy (DP) Dwork et al. (2006). DP allows to perform meaningful analyses of the entire dataset while protecting the privacy of individuals. It guarantees that if any single individual in a dataset were to change their data point, the (distribution of) outcomes of the differentially private mechanism remains roughly the same. The closeness of the outcomes is parametrized by a privacy parameter ε that captures the level of privacy. This ε represents the maximal privacy loss that any individual contributing data to the dataset is willing to accept, with small ε indicating high levels of privacy.

However, DP comes with a major limitation: It requires to set the privacy level ε uniformly for the entire dataset. Implicitly, this suggests that all individuals whose data is present in the dataset have the same privacy requirements. Yet, this is not accurate as individuals were shown to have diverse privacy requirements Jensen et al. (2005); Berendt et al. (2005); Acquisti & Grossklags (2005). By setting a uniform privacy budget in DP, this budget must match the individual whose privacy requirements are the strongest. This means that ε has to correspond to the highest privacy requirement in the dataset. Thereby—since the implementation of DP usually relies on the addition of noise with higher privacy requiring higher amounts of noise being added—DP often yields unfavorable privacy-accuracy trade-offs Li & Li (2009); Tramer & Boneh (2020); Bagdasaryan et al. (2019).

In this paper, we argue standard DP is overly conservative and propose a new algorithm to train ridge regression with per-individual personalized privacy guarantees.[1] We provide rigorous privacy

[1] In the remainder of this work, we will, without loss of generality, assume that each data point is contributed by a different individual.

proofs and accuracy guarantees for our algorithm, thereby, distinguishing ourselves from prior work on machine learning with personalized guarantees Alaggan et al. (2015); Jorgensen et al. (2015); Boenisch et al. (2023b;a) that solely provides empirical evaluations. Our personalized privacy techniques also differ from that of Boenisch et al. (2023b;a). A detailed discussion of related work is available in AppendixB.

In summary, we make the following contributions:

- We propose the first personalized DP algorithm specialized to the case of Ridge regression in Section 2, Algorithm (1).
- We provide rigorous privacy proofs in Section 2.1 and accuracy guarantees in Section 2.2.
- We perform extensive empirical evaluations in Section 3. We highlight that i) our algorithm significantly outperforms standard output perturbation-based DP ridge regression in terms of privacy-accuracy trade-offs on multiple datasets and diverse privacy distributions in Section 3.1 and Appendix D.1. We also show that we outperform the personalized privacy technique of Jorgensen et al. (2015) in Section 3.2 and Appendix D.2. Figure 1 plots the regularized test loss, while varying the fraction of high privacy data points (f_c), on the Medical cost dataset (Lantz, 2013). What we see is representative on all our experiments: the loss and standard deviation of our personalized privacy estimator is lower than that of Jorgensen et al. (2015).

Formally, the definition of personalized DP is the following:

Definition 1.1 (i-neighboring)**.** Two datasets D and D' are neighboring with respect to data point i (or "i-neighbors") if they differ only in data point i.

Definition 1.2 (Personalized DP)**.** A randomized algorithm $\mathcal{M}$ is ε_i-differentially private with respect to data point i, if for any outcome set $O \subset Range(\mathcal{M})$ and for all i-neighboring databases D, D'

$$\Pr[\mathcal{M}(D) \in O] \leq \exp(\varepsilon_i) \Pr[\mathcal{M}(D') \in O].$$

2 Algorithms and Guarantees for Personalized Privacy in Ridge Regression

Our Setup. Consider a dataset $\mathcal{D} = \{(x_i, y_i) \in \mathcal{X} \times \mathcal{Y} : i = 1, 2, \cdots, n\}$ consisting of a total n data points. We assume that features are bounded; without loss of generality, we work with $x_i \in [0, 1]^d$ for all $i \in [n]$. We also assume that the labels are bounded, and w.l.o.g. set $y_i \in [-1, 1]$ for all $i \in [n]$. Beyond this, each data point $i \in [n]$ has a *privacy requirement*, in the form of a DP parameter $\varepsilon_i > 0$. The lower the value of ε_i, the more stringent the privacy requirement of data point i, as per Definition 1.2.

Our main focus is Ridge regression. I.e., we are aiming to find a linear model $x^\top \bar{\theta}$, parametrized by $\bar{\theta}$, that predicts the labels as accurately as possible. Our goal is to find the $\bar{\theta}$ that minimizes the Ridge loss

$$L(\theta, \lambda) = \frac{1}{n} \sum_{i=1}^{n} \left(y_i - \theta^\top x_i\right)^2 + \lambda \|\theta\|_2^2.$$

However, we cannot release $\bar{\theta}$, as it encodes information about the dataset (x_i, y_i). Instead, we provide an estimator $\hat{\theta}$ whose performance on our Ridge loss is good, while at the same time ensuring that we satisfy DP with parameter ε_i for all data points $i \in [n]$ *simultaneously*.

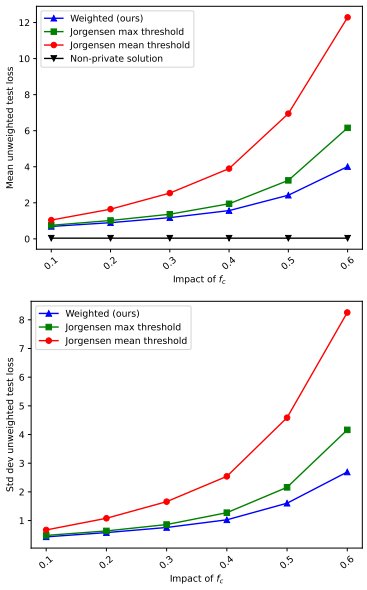


Figure 1: Mean (top) & std of the test loss (bottom) for our private estimator (in blue) vs Jorgensen et al. (2015) while varying f_c, regularization parameter $\lambda = 0.7$.

Our Main Algorithm. The main idea of our algorithm, "Personalized-Differentially-Private Output Perturbation" (or "PDP-OP") is as follows: if we

wanted to obtain traditional, non-personalized DP, we could first compute a non-private estimate $\bar{\theta}$, then add well-chosen noise Z for privacy with density $\nu(b) \propto \exp(-\eta\|b\|_2)$. This takes inspiration from Chaudhuri & Monteleoni (2008); Chaudhuri et al. (2011); however, our analysis exhibits differences due to differences in our estimators to incorporate personalized DP. Here, to instead obtain personalized DP, we rely on an idea studied by Cummings & Durfee (2020): manipulating or pre-processing the sensitivity (i.e., how much a given data point can change the outcome) of our query with respect to each data point i. More precisely, our sensitivity pre-processing technique re-weights each data point i by a weight w_i. The smaller the weight w_i, the smaller the impact the i-th data point has on the output model. Intuitively, this means that lower w_i's correspond to less information encoded in the output $\hat{\theta}$ about data point i, and better privacy for i. When giving different w_i to different data points, we can then ensure different, personalized privacy levels for different data points. Our algorithm is given formally in Algorithm 1. We then show formally in Section 2.1 how to choose the weights w_i and the noise parameter η in order to guarantee personalized DP with respect to privacy preferences $\varepsilon_1, \ldots, \varepsilon_n$.

2.1 Privacy Guarantees

We now state the personalized privacy guarantee obtained by our algorithm. We remind the reader that this privacy guarantee relies on the observations being normalized such that $x_i \in [0,1]^d$ and $y_i \in [-1,1]$ for all $i \in [n]$:

Theorem 2.1. *Fix privacy specifications $\varepsilon_1, \ldots, \varepsilon_n > 0$. Let $B(\lambda) = \min\left(\frac{1}{\sqrt{\lambda}}, \frac{\sqrt{d}}{\lambda}\right)$. Algorithm 1 with parameters $w_i = \frac{\varepsilon_i}{\sum_{j=1}^n \varepsilon_j}$ for all i and $\eta = \frac{\lambda}{2\sqrt{d}\left(1+\sqrt{d}B(\lambda)\right)} \sum_{j=1}^n \varepsilon_j$ is ε_i-personalized differentially private with respect to data point i for every $i \in [n]$.*

Additionally, we show another version of our privacy guarantee that uses additional assumptions on the data and on the best regression parameter absent regularization, when such information is available:

Assumption 2.2 (Bounded $\bar{\theta}$)**.** Let $\bar{\theta}_0 = \arg\min_\theta \sum_{i=1}^n w_i \left(y_i - \theta^\top x_i\right)^2$, when $\lambda = 0$. There is a known constant B such that $\left\|\bar{\theta}_0\right\|_2 \leq B$.

We incorporate such boundedness assumptions as they have also been used in previous work, such as Wang (2018); Arora et al. (2022). Our privacy guarantee, under additional Assumption 2.2, is given by:

Theorem 2.3. *Suppose Assumption 2.2 holds. Fix privacy specifications $\varepsilon_1, \ldots, \varepsilon_n > 0$. Algorithm 1 with parameters $w_i = \frac{\varepsilon_i}{\sum_{j=1}^n \varepsilon_j}$ for all i and $\eta = \frac{\lambda}{2\sqrt{d}(B\sqrt{d}+1)} \sum_{j=1}^n \varepsilon_j$ is ε_i-personalized differentially private with respect to data point i for every $i \in [n]$.*

We can easily see that if $B \leq B(\lambda) = \min\left(\frac{1}{\sqrt{\lambda}}, \frac{\sqrt{d}}{\lambda}\right)$, this bound adds noise Z with a bigger parameter η compared to in Theorem 2.1, which corresponds to adding *less* noise. This will lead to better accuracy guarantees in regimes in which we put little weight on the regularization parameter, provided that we know or can estimate such a bound B. The proofs for Theorem 2.1 and 2.3 are provided in Appendix C.1

2.2 Accuracy Guarantees

We now provide theoretical bounds on the accuracy of our framework. We note that we are the first to provide such theoretical accuracy bounds for ridge regression with personalized DP. The framework of Jorgensen et al. (2015) is relatively general, but said generality prevents them from obtaining worst-case theoretical accuracy bounds, and they focus on an empirical evaluation of the performance (in terms of loss or accuracy) of their sampling framework.

We make the assumption that the label generating process is in fact approximately linear, which is the main use case in which linear regression should be used in the first place (Vershynin, 2018). Importantly, note that this assumption is only made in order to characterize the theoretical accuracy of our framework. Our personalized DP bounds of Section 2.1 crucially *do not* rely on Assumption 2.4.

Assumption 2.4. Given a feature vector x_i, the label y_i is given by $y_i = x_i^\top \theta^* + Z_i$ where $\theta^* \in \mathbb{R}^d$ and the Z_i's are independent and identically distributed Gaussian variables with mean 0 and standard deviation $\sigma > 0$.

We now provide a bound on how well we recover θ^*, the true data generating process, as closely as possible as a function of our dataset and our choice of privacy parameters.

Theorem 2.5 (Accuracy of $\hat{\theta}$)**.** *Let η be chosen as per Theorems 2.1 absent assumptions, and Theorem 2.3 under Assumption 2.2. For any $\delta > 0$, with probability at least $1 - \delta$, we have that for any $\lambda > 0$,*

$$\|\theta^* - \hat{\theta}\|_2 \leq \frac{\|\theta^*\|}{1 + \frac{\lambda_{min}\left(\sum_{i=1}^n w_i x_i x_i^\top\right)}{\lambda}} + \frac{1}{\eta}\left(d + \sqrt{\frac{2d}{\delta}}\right) + \frac{\sigma}{\lambda}\sqrt{\frac{2d}{\delta}}\|\vec{w}\|$$

for all $\lambda > 0$, where $\vec{w} = (w_1, \ldots, w_n)$.

The proof of Theorem 2.5 is in Appendix C.2. The interpretation is as follows: the distance between our private estimate $\hat{\theta}$ and the true data generating parameter θ^* is upper bounded by three terms, the first is a bias term from ridge regresssion, $\|\theta^*\| \Big/ \left(1 + \lambda_{min}\left(\sum_{i=1}^n w_i x_i x_i^\top\right)/\lambda\right)$. If the problem is well conditioned [2] then as $\lambda \to 0$ this first term vanishes. If the problem is not well conditioned then the bias from ridge regression is unavoidable. The second term $\frac{1}{\eta}\left(d + \sqrt{\frac{2d}{\delta}}\right)$, is due to the noise added for privacy with parameter $\eta \propto \sum_{j=1}^n \varepsilon_j$. Absent personalization, $\eta \propto n \min_j \varepsilon_j$, which leads to significantly more noise addition for privacy. Further, in the second term, η is an increasing function of λ, thus the second term is therefore decreasing in λ. This is in contrast with the bias term, increasing λ increases the weight on the regularization hence increases the bias of our estimator; however, at the same time, it decreases the amount of noise we need to add for privacy. Finally, the third term $\frac{\sigma}{\lambda}\sqrt{\frac{2d}{\delta}}\|\vec{w}\|$ captures noise in the labels themselves.

3 EXPERIMENTS

In this section, we evaluate the performance of our algorithm experimentally. Importantly, we highlight that our goal *is not* to evaluate the performance of "output perturbation" (which first computes a non-private estimator then adds noise for privacy) for private regression, as this has been done extensively in previous work Chaudhuri & Monteleoni (2008); Chaudhuri et al. (2011). Rather, we highlight the performance of our re-weighting technique, in particular compared to the absence of data reweighting (which gives the same level of privacy to all data points) and to the sampling-based technique for personalized privacy of Jorgensen et al. (2015). For this reason and to isolate the performance of our re-weighting technique versus the sampling of Jorgensen et al. (2015), we fix our experimental evaluation to have all baselines be based on output perturbation techniques.

We divide this section into two parts: i) we show how much the addition of personalized privacy improves accuracy compared to non-personalized privacy; ii) we compare our results to Jorgensen et al. (2015), and show improvement both in terms of accuracy and variance of the estimator.

Choice of Privacy Budgets. To validate our personalized privacy setting, we follow a similar segregation scheme as Alaggan et al. (2015); Jorgensen et al. (2015). We categorize data points into 3 segments, in order of most stringent to less stringent privacy requirements: *conservatives* (high privacy), *mediums* or *pragmatists* (medium privacy), and *liberals* (low privacy). The fraction of conservatives, mediums, and liberals in the population is denoted by f_c, f_m and $f_l = 1 - (f_c + f_m)$ respectively. Each segment has their own privacy parameter denoted respectively ε_c, ε_m and ε_l, where $\varepsilon_c < \varepsilon_m < \varepsilon_l$ (remember that lower ε, means stronger privacy requirement). In our experiments, we assign the personalized privacy budgets to the conservatives and mediums by uniformly sampling from the ranges $[\varepsilon_c, \varepsilon_m]$, $[\varepsilon_m, \varepsilon_l]$ respectively. As for the liberals, they all receive the same single highest privacy budget ε_l. This follows Niu et al. (2020); Jorgensen et al. (2015).

[2] i.e., $\lambda_{min}\left(\sum_{i=1}^n w_i x_i x_i^\top\right) > 0$

As default values, we set $f_c = 0.34, f_m = 0.43, f_l = 0.23$, $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0$ unless otherwise specified. We provide experiments in Appendix D where we show how our insights extend as we change these privacy parameters.

Synthetic Data Generation. For the synthetic data, we draw θ^* uniformly over the unit sphere S^{d-1}. Each feature x is drawn uniformly over support $[0, 1]^d$, and its corresponding label $y = \frac{x^\top \theta^*}{\sqrt{d}}$: the label then satisfies $|y| \leq \frac{1}{\sqrt{d}} \|x\|_2 \|\theta^*\|_2 \leq 1$. We consider perfect linear relationships between x and y with no noise in the labels (i.e., $\sigma = 0$ as per the notations of Section 2.2); we do so to decouple the effect of linear models being an imperfect hypothesis class from the performance of our method. We rely on our real dataset described below, both in the main body and in Appendix D, for situations in which the relationship between features and labels can only approximately be captured by a linear model.

Real Dataset. For our experiments on real data, we use the "Medical Cost" dataset (Lantz, 2013) which looks at an individual medical cost prediction task. Each individual's features are split into three numeric {`age, BMI, #children`} and three categorical features {`sex, smoker, region`}. The dataset also has a real-valued medical `charges` column that we use as our label. **Data pre-processing.** We use min-max scaling to normalize the numeric features as well as the label to the range $[0, 1]$. For any categorical features, we use standard one-hot-encoding. To deal with affine rather than simply linear relationships in the data, we add a $d + 1$-th feature to each feature vector x, that corresponds to our model's intercept.

Metrics. We evaluate the performance of $\hat{\theta}$ on a held-out test set of size N_{test}, using the following metrics : (1) **Unregularized test loss:** $\sum_{i=1}^{N_{test}} \frac{1}{N_{test}} (y_i - x_i^\top \hat{\theta})^2$ and (2) **Regularized test loss:** $\sum_{i=1}^{N_{test}} \frac{1}{N_{test}} (y_i - x_i^\top \hat{\theta})^2 + \lambda \left\| \hat{\theta} \right\|_2^2$. The λ we use in our evaluation is the same λ that we use in our training loss and in Algorithm 1.

3.1 Improvements over standard Differential Privacy

Our first results show the improvements in accuracy when we use personalized DP as opposed to standard (or non-personalized) DP. To do so, we compare to what we call the *non-personalized* baseline which provides the same privacy level ε to all data points. We let $\varepsilon^* = \min_i \varepsilon_i$ is chosen to satisfy the most stringent privacy preferences (remember that smaller ε means more privacy) among all data points. To implement this baseline, we simply use Algorithm 1, but with the privacy preference profile being $(\varepsilon^*, \ldots, \varepsilon^*)$. Note that the weights are all the same and equal to $1/n$ and the added noise scales as a function of ε^*. Hence our baseline implementation just follows the standard regression algorithm with output perturbation of Chaudhuri & Monteleoni (2008) and Chaudhuri et al. (2011).

Table 1 shows the performance of our algorithm versus the non-personalized baseline across varying regularization λ, while fixing the other parameters $f_c, f_m, \varepsilon_c, \varepsilon_m$. We note that we get consistent improvements of several order of magnitudes across all values of λ. The improvement is, for example, roughly of an order of magnitude of 100 when it comes to both unregularized and regularized mean-squared error on the test set. This shows that leveraging differing privacy preferences across differing data points can lead to huge improvements in terms of privacy-accuracy trade-offs for differential privacy. Table 2 shows that significant improvements also occur on the real dataset. Appendix D.1 provides more experiments where we vary parameters $f_c, \varepsilon_c, \varepsilon_m, n$ on the synthetic and real datasets.

3.2 Comparison to Jorgensen et al. (2015)

Experimental setup. We compare our approach to that of Jorgensen et al. (2015), who proposed the first algorithm for personalized differential privacy in the central privacy model. Their approach is the following: first, they pick a threshold t. Then, they sample each data point i in D with probability $\frac{\exp\{\varepsilon_i\}-1}{\exp\{t\}-1}$ if $\varepsilon_i < t$, and probability 1 otherwise. Lastly, they run a standard non-personalized algorithm that is t-differentially private. We fix the non-personalized algorithm to follow the output perturbation technique described in Chaudhuri & Monteleoni (2008); Chaudhuri

et al. (2011) to keep comparisons apple-to-apple and only compare the impact of our re-weighting versus the thresholding then sub-sampling approach of Jorgensen et al. (2015). We implement the non-personalized estimator as per the baseline described in Section 3.1. For the choice of t, we try out both $t = \max_i \varepsilon_i$ (that we refer to as "Jorgensen max" or "max threshold") and $t = \frac{1}{n}\sum_i \varepsilon_i$ (that refer to as "Jorgensen mean" or "mean threshold"). Both these choices of t were proposed by Jorgensen et al. (2015) itself.

In the rest of this section, we present our experimental results. We note that our framework consistently leads to improved performance over Jorgensen et al. (2015). This is seen in two ways: first, our unregularized and regularized losses are consistently lower than those of Jorgensen for the vast majority of choices of instance parameters and of regularization parameter λ. This show that our re-weighting method is consistently more accurate compared to the subsampling method of Jorgensen et al. (2015) when it comes to accuracy. Beyond this, we also note that our results are more consistent: the standard deviation of the loss of our technique is also lower than that of Jorgensen et al. (2015). I.e., our results are more consistent across different runs of the algorithms and different realizations of the noise.

Intuitively, one of the advantages of our technique over that of Jorgensen et al. (2015) is that by re-weighting instead of sampling, we do not discard any of our dataset; we believe this is one potential source of improvement. Another, more subtle reason, may be that our framework adds all noise Z centrally, at the end of the computation, while Jorgensen et al. (2015) adds noise both locally (when sub-sampling data points) and centrally. It is well understood that adding noise centrally (i.e., within the computation) in differential privacy leads to better privacy-accuracy trade-offs than adding noise locally (i.e., at the level of each data point), which may be another reason for our improvements.

Improvements in loss. Table 11 provides a snapshot of our performance versus that of the baseline of Jorgensen et al. (2015), for our default choice of privacy parameters. We perform consistently better than Jorgensen et al. (2015) across both our metrics (unregularized and regularized loss), with improvements in loss of up to roughly 20 percent under the max threshold. Jorgensen et al. (2015)'s results when using the mean threshold instead are consistently worse. Table 12 shows that similar insights hold on our real dataset.

Improvements in variability of the results. Figure 10 and 11 show the standard deviation of our loss compared to that of Jorgensen et al. (2015), estimated across $10,000$ runs for each technique on the synthetic and real datasets respectively. The figure clearly highlights how our method exhibits less variability, leading to more consistent results across different runs and realizations of the noise.

We also provide additional experiments where we change the parameters of the problem such as $f_c,\ \varepsilon_c,\ \varepsilon_m,\ n$ and when we consider our real dataset in Appendix D.2, and note there that our insights still hold across these experiments.

4 Conclusion and Future Work

We proposed a new algorithm, Personalized-DP Output Perturbation (PDP-OP), which allows to train Ridge regression models with individual per-data point privacy requirements. We formally prove PDP-OP's personalized privacy guarantees and provide rigorous and theoretical results for the accuracy guarantees of our framework. We are in fact the first to provide a theoretical accuracy guarantee for personalized-DP methods in machine learning, to the best of our knowledge. Our empirical evaluation on synthetic and real datasets highlights that PDP-OP significantly outperforms non-personalized DP, highlighting the need for personalized DP to vastly improve privacy-accuracy trade-offs in private ML. We also show that we outperform previous techniques for personalized DP, showing the advantages of using re-weighting over sub-sampling techniques.

The current paper aims to provide initial algorithms for personalized DP tailored to the case of Ridge regression. We chose to use *output perturbation* as a simple starting point to provide initial insights and algorithms into personalized DP, and the benefits of data re-weighting over data sub-sampling. We, however, believe that there is still some leeway to improve the privacy-accuracy trade-offs of linear regression with personalized DP. In future work, we will incorporate our re-weighting technique with more advanced techniques for private regression, such as *objective perturbation*, *summary statistic perturbation*, or *private gradient descent* (see Appendix B for more details and examples).

REFERENCES

A. Acquisti and J. Grossklags. Privacy and rationality in individual decision making. *IEEE Security & Privacy*, 3(1):26–33, 2005. doi: 10.1109/MSP.2005.22.

Daniel Alabi, Audra McMillan, Jayshree Sarathy, Adam Smith, and Salil Vadhan. Differentially private simple linear regression. *Proceedings on Privacy Enhancing Technologies*, 2022(2):184–204.

Mohammad Alaggan, Sébastien Gambs, and Anne-Marie Kermarrec. Heterogeneous differential privacy. *arXiv preprint arXiv:1504.06998*, 2015.

Raman Arora, Raef Bassily, Cristóbal Guzmán, Michael Menart, and Enayat Ullah. Differentially private generalized linear models revisited. *Advances in Neural Information Processing Systems*, 35:22505–22517, 2022.

Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. Differential privacy has disparate impact on model accuracy. *Advances in neural information processing systems*, 32, 2019.

Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th annual symposium on foundations of computer science*, pp. 464–473. IEEE, 2014.

Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Guha Thakurta. Private stochastic convex optimization with optimal rates. *Advances in neural information processing systems*, 32, 2019.

Bettina Berendt, Oliver Günther, and Sarah Spiekermann. Privacy in e-commerce: Stated preferences vs. actual behavior. *Communications of the ACM*, 48(4):101–106, 2005.

Franziska Boenisch, Christopher Mühl, Adam Dziedzic, Roy Rinberg, and Nicolas Papernot. Have it your way: Individualized privacy assignment for dp-sgd. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a. URL https://openreview.net/forum?id=XXPzBhOs4f.

Franziska Boenisch, Christopher Mühl, Roy Rinberg, Jannis Ihrig, and Adam Dziedzic. Individualized pate: Differentially private machine learning with individual privacy guarantees. In *23rd Privacy Enhancing Technologies Symposium (PoPETs)*, 2023b.

T Tony Cai, Yichen Wang, and Linjun Zhang. The cost of privacy: Optimal rates of convergence for parameter estimation with differential privacy. *The Annals of Statistics*, 49(5):2825–2850, 2021.

Kamalika Chaudhuri and Claire Monteleoni. Privacy-preserving logistic regression. *Advances in neural information processing systems*, 21, 2008.

Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(3), 2011.

Xiangyi Chen, Steven Z Wu, and Mingyi Hong. Understanding gradient clipping in private sgd: A geometric perspective. *Advances in Neural Information Processing Systems*, 33:13773–13782, 2020.

Rachel Cummings and David Durfee. Individual sensitivity preprocessing for data privacy. In *Proceedings of the Fourteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 528–547. SIAM, 2020.

Rachel Cummings, Katrina Ligett, Aaron Roth, Zhiwei Steven Wu, and Juba Ziani. Accuracy for sale: Aggregating data with a variance constraint. In *Proceedings of the 2015 conference on innovations in theoretical computer science*, pp. 317–324, 2015.

Rachel Cummings, Hadi Elzayn, Vasilis Gkatzelis, Emmanouil Pountourakis, and Juba Ziani. Optimal data acquisition with privacy-aware agents. In *First IEEE Conference on Secure and Trustworthy Machine Learning*, 2023.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pp. 265–284. Springer, 2006.

Alireza Fallah, Ali Makhdoumi, Azarakhsh Malekian, and Asuman Ozdaglar. Optimal and differentially private data acquisition: Central and local mechanisms. In *Proceedings of the 2022 ACM Conference on Economics and Computation*, pp. 1141, 2022.

Anna C. Gilbert and Audra McMillan. Property testing for differential privacy. *2018 56th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 249–258, 2018. URL https://api.semanticscholar.org/CorpusID:49295618.

Prateek Jain and Abhradeep Guha Thakurta. (near) dimension independent risk bounds for differentially private learning. In *International Conference on Machine Learning*, pp. 476–484. PMLR, 2014.

Carlos Jensen, Colin Potts, and Christian Jensen. Privacy practices of internet users: Self-reports versus observed behavior. *International Journal of Human-Computer Studies*, 63(1-2):203–227, 2005.

Zach Jorgensen, Ting Yu, and Graham Cormode. Conservative or liberal? personalized differential privacy. In *2015 IEEE 31st International Conference on Data Engineering*, pp. 1023–1034, 2015. doi: 10.1109/ICDE.2015.7113353.

Daniel Kifer, Adam Smith, and Abhradeep Thakurta. Private convex empirical risk minimization and high-dimensional regression. In *Conference on Learning Theory*, pp. 25–1. JMLR Workshop and Conference Proceedings, 2012.

Brett Lantz. Medical Cost Personal Datasets — kaggle.com, 2013. URL https://www.kaggle.com/datasets/mirichoi0218/insurance.

Haoran Li, Li Xiong, Zhanglong Ji, and Xiaoqian Jiang. Partitioning-based mechanisms under personalized differential privacy. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 615–627. Springer, 2017.

Tiancheng Li and Ninghui Li. On the tradeoff between privacy and utility in data publishing. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 517–526, 2009.

Ben Niu, Yahong Chen, Boyang Wang, Jin Cao, and Fenghua Li. Utility-aware exponential mechanism for personalized differential privacy. In *2020 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, 2020. doi: 10.1109/WCNC45663.2020.9120532.

Ben Niu, Yahong Chen, Boyang Wang, Zhibo Wang, Fenghua Li, and Jin Cao. Adapdp: Adaptive personalized differential privacy. In *IEEE INFOCOM 2021-IEEE Conference on Computer Communications*, pp. 1–10. IEEE, 2021.

Shuang Song, Om Thakkar, and Abhradeep Thakurta. Characterizing private clipped gradient descent on convex generalized linear problems. *arXiv preprint arXiv:2006.06783*, 2020.

Shuang Song, Thomas Steinke, Om Thakkar, and Abhradeep Thakurta. Evading the curse of dimensionality in unconstrained private glms. In *International Conference on Artificial Intelligence and Statistics*, pp. 2638–2646. PMLR, 2021.

Florian Tramer and Dan Boneh. Differentially private learning needs better features (or much more data). In *International Conference on Learning Representations*, 2020.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

Yu-Xiang Wang. Revisiting differentially private linear regression: optimal and adaptive prediction & estimation in unbounded domain. *arXiv preprint arXiv:1803.02596*, 2018.

Algorithm 1 Personalized-Differentially-Private Output Perturbation (PDP-OP)

Inputs: Dataset $\mathcal{D} = \{(x_i, y_i) \text{ for } i \in [n]\}$; weights vector $\mathbf{w} \geq 0$ with $\sum_{i=1}^{n} w_i = 1$; noise parameter η.

Output: Private estimator $\hat{\theta}$

1: Compute non-private estimate $\bar{\theta}$ as follows:

$$\bar{\theta} = \arg\min_{\theta} \sum_{i=1}^{n} w_i \left(y_i - \theta^\top x_i\right)^2 + \lambda \|\theta\|_2^2. \tag{1}$$

2: Sample Z as a random variable with probability density function $\propto \exp(-\eta \|b\|_2)$

3: Return private estimate $\hat{\theta} = \bar{\theta} + Z$

Remark .1 (How to sample Z). It is known that for probability density function $\nu(b) \propto \exp(-\eta \|b\|_2)$ for Z:

- $\|Z\|_2$ follows a Gamma(α, β) distribution with $\alpha = d$ and $\beta = \eta$, which has density $f(r) \propto r^{d-1} \exp\{-\eta r\}$.[3]
- For any given value $r \triangleq \|Z\|_2$, Z is uniform on the ℓ_2-sphere of radius r. This can be seen immediately as the density only depends on $\|Z\|$, so all realizations z with the same norm have the same density.

In turn, to sample Z, it suffices to i) sample the radius R from Gamma(d, η), then ii) sample Y uniformly at random from the ℓ_2-sphere of radius 1, and set $Z = RY$.

A A Potential Limitation: the Privacy of Privacy Costs

In certain sensitive applications, an individual's choice of privacy budget can reflect some sensitive information. Imagine a medical context with a dataset that consists of individuals that do and individuals that do not have a rare disease. The latter ones might prefer higher privacy protection to hide their condition. If an attacker with access to the trained model was able to deduce the individual's privacy budget, they might, in turn, be able to draw conclusions on the individual's medical state. This provides a point of attack towards our privacy guarantees.

We note, however, that deducing privacy budgets is generally a hard problem, alleviating our concerns when it comes to data–privacy budget correlations. Theoretical results from ML, e.g., Gilbert & McMillan (2018), have shown that one cannot currently perform sample-efficient black-box audits to determine the privacy-budget of a trained model.

B Related work

Personalized Privacy. Personalizing privacy guarantees are highly relevant, given that studies showed how society consists at least of three different groups of individuals, requiring strong, average, or weak privacy protection Jensen et al. (2005); Berendt et al. (2005); Acquisti & Grossklags (2005). Without personalization, when applying DP to datasets that hold data from individuals with different privacy requirements, ε needs to be set to the lowest ε encountered among all individuals whose data we choose to use in our computation. This can often yield unfavorable privacy-utility trade-offs, due to having to throw away too much data or to have to use a stringent value of ε for everyone. Instead, several previous works concurrently introduced the concept of *personalized* DP. It was introduced formally in Alaggan et al. (2015) and Jorgensen et al. (2015) in the context of

[3] An easy way to see this informally is the following: the total mass on $\|Z\|_2 = r$ is proportional to the total mass on all Z's with norm r—which is proportional to $\frac{2\pi^{d/2}}{\Gamma(d/2)} r^{d-1}$ (the surface area of the d-dimensional ℓ_2-sphere)—multiplied by the mass on any single Z of norm r—which is proportional to $\exp\{-\eta r\}$. A more formal proof, omitted here, consists in writing the integration for $P[\|Z\|_2 \leq R]$, then doing a change of variable to hyper-spherical coordinates in the corresponding integral.

central DP; it was also used informally by Cummings et al. (2015) in the context of mechanism design for data acquisition with local DP.

Two of the most prevalent techniques for personalized DP are personalized *data sampling* and *sensitivity pre-processing*.

Both methods change how much the output of a computation depends on a particular data point: the lesser the dependency on a given data point, the more privacy this point gets. The idea of data sampling for personalized privacy was introduced by Jorgensen et al. (2015), and later used in the works of Niu et al. (2021); Boenisch et al. (2023b;a). Data sampling introduces randomness in the dataset: each data point can be sub-sampled or up-sampled before being fed into a standard DP algorithm.

Sensitivity pre-processing, which is the approach used in this work, in contrast, can be implemented deterministically. It modifies the query of interest to have different sensitivities for different data points, where sensitivity is a standard notion in DP on how much a computation can change across neighbouring datasets.

Sensitivity pre-processing for personalized privacy was originally introduced by Alaggan et al. (2015) through linear pre-processing or "stretching" of the input data; a caveat of this method is that it requires strong assumptions on how changing the data changes per-user sensitivity. A general-purpose method for manipulating the sensitivity of a query while maintaining accuracy is provided by Cummings & Durfee (2020), but is unfortunately NP-hard to implement in the general case and is constructive, rather than given in closed form. Specializations of this method for the case of moment estimation have been recently used in Fallah et al. (2022); Cummings et al. (2023): both papers rely on weighted moment estimation, where the weight is lower for data points with stronger privacy requirements. A difficulty with such re-weightings is that they often need to be tailored to the specific learning task at hand. This is the approach we take and challenge we face in this work. Finally, Li et al. (2017) proposed two partitioning algorithms that first separate the data in different groups according to privacy requirements, and then process these groups separately. This approach was shown sub-optimal for learning-based applications Boenisch et al. (2023b).

Private Empirical Risk Minimization. There has been a significant line of work on making linear regression, generalized linear models, and empirical risk minimization (ERM) differentially-private (Chaudhuri & Monteleoni, 2008; Chaudhuri et al., 2011; Kifer et al., 2012; Bassily et al., 2014; Jain & Thakurta, 2014; Wang, 2018; Bassily et al., 2019; Chen et al., 2020; Song et al., 2020; Cai et al., 2021; Song et al., 2021; Alabi et al.; Arora et al., 2022). These papers focus on standard DP, as opposed to personalized DP, and span a relatively large number of different techniques. Most relevant to us are the initial works in this space by Chaudhuri & Monteleoni (2008); Chaudhuri et al. (2011). In particular, they analyze obtaining DP in regression and ERM through an *output perturbation* technique: ERM is first performed non-privately, then noise is added directly to the non-private estimator. We similarly rely on output perturbation in this paper, noting that this is a good starting point to the study of personalized DP in the context of regression. Most recently, output perturbation saw a new analysis for generalized linear models by Arora et al. (2022).

C Proof of Main Results

C.1 Full Proof of Theorems 2.1 and 2.3

Preliminaries: Bound on $\|\bar{\theta}\|$ for Algorithm 1 First, we show that the norm of $\bar{\theta}$ is bounded for Algorithm 1. This bound will be useful in bounding the gradient difference across two neighbouring databases in the proof of the privacy guarantee of Algorithm 1.

Lemma C.1. *For any $\sum_{i=1}^{n} w_i = 1, w_i \geq 0$, the unconstrained minimizer of* (1)*, $\bar{\theta}$, satisfies:*

$$\left\|\bar{\theta}\right\|_2 \leq \frac{1}{\sqrt{\lambda}}.$$

Proof. Let the weighted loss be defined as:

$$L_w(\theta) \triangleq \sum_{i=1}^{n} w_i \left(y_i - \theta^\top x_i\right)^2 + \lambda \|\theta\|_2^2.$$

For any $\theta \in \mathbb{R}^d$, $L_w(\bar{\theta}) \leq L_w(\theta)$. Therefore,

$$\lambda \left\|\bar{\theta}\right\|^2 \leq L_w(\bar{\theta}) \leq L_w(0) = \sum_{i=1}^n w_i y_i^2 \leq 1.$$

□

Lemma C.2. *For any $\sum_{i=1}^n w_i = 1, w_i \geq 0$, the unconstrained minimizer of (1), $\bar{\theta}$, satisfies $\left\|\bar{\theta}\right\|_2 \leq \frac{\sqrt{d}}{\lambda}$.*

Proof. We first find the closed form for $\bar{\theta}$. $\bar{\theta}$ is simply the unique solution to the unconstrained minimization problem

$$\arg\min_{\theta \in \mathbb{R}^d} \sum_{i=1}^n w_i \left(y_i - \theta^\top x_i\right)^2 + \lambda \|\theta\|_2^2.$$

Therefore,

$$\bar{\theta} = \arg\min_{\theta \in \mathbb{R}^d} \sum_{i=1}^n w_i \left(y_i - \theta^\top x_i\right)^2 + \lambda \|\theta\|_2^2.$$

Taking the first order-condition, we note that we must have

$$-2 \sum_{i=1}^n w_i x_i \left(y_i - x_i^\top \bar{\theta}\right) + 2\lambda\theta = 0.$$

This can be rewritten as

$$2 \left(\sum_{i=1}^n w_i x_i x_i^\top + \lambda I \right) \bar{\theta} = 2 \sum_{i=1}^n w_i x_i y_i.$$

Because $\lambda > 0$ and $\sum_{i=1}^n w_i x_i x_i^\top$ is positive semi-definite, $\sum_{i=1}^n w_i x_i x_i^\top + \lambda I$ is invertible, and we obtain the following closed-form expression for $\bar{\theta}$:

$$\bar{\theta} = \left(\sum_{i=1}^n w_i x_i x_i^\top + \lambda I \right)^{-1} \sum_{i=1}^n w_i x_i y_i.$$

Taking the ℓ_2-norm of both sides, we obtain, letting $\lambda_{min}(M),\ \lambda_{max}(M)$ respectively denote the lowest and highest eigenvalues of any given matrix M:

$$\left\|\bar{\theta}\right\|_2 \leq \lambda_{max} \left(\left(\lambda I + \sum_{i=1}^n w_i x_i x_i^\top \right)^{-1} \right) \left\| \sum_{i=1}^n w_i y_i x_i \right\|_2 \tag{2}$$

$$\leq \lambda_{max} \left(\left(\lambda I + \sum_{i=1}^n w_i x_i x_i^\top \right)^{-1} \right) \cdot \sqrt{d} \tag{3}$$

$$\leq \frac{1}{\lambda_{min} \left(\lambda I + \sum_{i=1}^n w_i x_i x_i^\top \right)} \cdot \sqrt{d} \tag{4}$$

$$\leq \frac{\sqrt{d}}{\lambda}, \tag{5}$$

where the second inequality follows from $\|x_i\|_2 \leq \sqrt{d}$ □

Under additional Assumption 2.2, note that $\left\|\bar{\theta}\right\| \leq \left\|\bar{\theta}_0\right\|$: indeed, if not, $\bar{\theta}_0$ is a better solution to the Ridge problem with parameter λ, noting that it has both i) lower—and in fact optimal—unregularized loss and ii) lower ℓ_2-penalization. Then, we have $\|\bar{\theta}\| \leq B$. In the rest of the proof, we let $B(\lambda) \triangleq \min\left(\frac{1}{\sqrt{\lambda}}, \frac{\sqrt{d}}{\lambda}\right)$ absent assumptions, and $B(\lambda) \triangleq B$ under Assumption 2.2. Now, in both cases, we have $\left\|\bar{\theta}\right\| \leq B(\lambda)$.

Sensitivity analysis of $\bar{\theta}$ through strong convexity and bounded gradients We start with the following lemma that will help us bound the gradient different over two minimization problem: one over loss function $G(\theta)$ and one over modified loss function $G(\theta) + g(\theta)$, corresponding to the losses on neighboring databases X' and X. The lemma is a slight modification from Lemma 7 of Chaudhuri et al. (2011) that works with the gradient of $g(\theta)$ evaluated at a specific well-chosen point, instead of the maximum norm for the gradient of $g(\theta)$ over $\mathbb{R}^d$

Lemma C.3. *Let $G(\theta)$ and $g(\theta)$ be two vector valued, continuous, differentiable functions with $G(\theta)$ and $G(\theta)+g(\theta)$ both γ-strongly convex. Let $\theta_1 = \arg\min_\theta G(\theta)+g(\theta)$, $\theta_2 = \arg\min_\theta G(\theta)$, then*

$$\|\theta_1 - \theta_2\|_2 \leq \frac{1}{\gamma}\|\nabla g(\theta_1)\|_2.$$

Proof. Note that θ_1 and θ_2 must satisfy the first order conditions, i.e.

$$\nabla G(\theta_2) = 0 = \nabla G(\theta_1) + \nabla g(\theta_1). \tag{6}$$

Further, by strong convexity of G with parameter γ, we have

$$\begin{aligned}\gamma\|\theta_1 - \theta_2\|^2 &\leq \left(\nabla G(\theta_2) - \nabla G(\theta_1)\right)^\top (\theta_2 - \theta_1) \\ &= \nabla g(\theta_1)^\top(\theta_2 - \theta_1) \\ &\leq \|\nabla g(\theta_1)\| \cdot \|\theta_1 - \theta_2\|,\end{aligned}$$

where line 2 follows from Equation (6) and line 3 follows from Cauchy-Schwarz. □

This allows us to bound the sensitivity with respect to agent i of the non-private Ridge regression minimizer $\bar{\theta}$ as a function of w_i. Note that we define the ℓ_2 sensitivity of $\bar{\theta}$ with respect to agent i as

$$\Delta_i\bar{\theta} = \max_{D,D' \text{ i-neighboring}} \|\bar{\theta}(D) - \bar{\theta}(D')\|_2,$$

where $\bar{\theta}(D) = \arg\min_\theta \sum_{i=1}^n w_i(y_i - \theta^\top x_i)^2 + \lambda\|\theta\|_2^2$. The sensitivity bound is then given by:

Theorem C.4. *Let $\bar{\theta} = \arg\min_{\theta\in\mathbb{R}^d} \sum_{i=1}^n w_i\left(y_i - \theta^\top x_i\right)^2 + \lambda\|\theta\|_2^2$. $\bar{\theta}$ has ℓ_2-sensitivity $(\Delta\theta)_i$ with respect to agent i that satisfies $\Delta_i\bar{\theta} \leq \frac{2\sqrt{d}w_i}{\lambda}\left(\sqrt{d}B(\lambda) + 1\right)$.*

Proof. Consider two neighbouring databases $X = ((x_1, y_1), \ldots, (x_n, y_n))$ and $X = ((x_1, y_1), \ldots, (x_i', y_i'), \ldots, (x_n, y_n))$ that only differ in the data of agent i. Let $G(\theta) + g(\theta) = \sum_{j=1}^n w_j\left(y_j - \theta^\top x_j\right)^2 + \lambda\|\theta\|_2^2$, and $g(\theta) = w_i\left(y_i - \theta^\top x_i\right)^2 - w_i\left(y_i' - \theta^\top x_i'\right)^2$.

First, we note that

$$G(\theta) + g(\theta) = \sum_j w_j\left(y_j - \theta^\top x_j\right)^2 + \lambda\|\theta\|_2^2$$

is the Ridge loss on database D, while

$$\begin{aligned}G(\theta) &= \sum_j w_j\left(y_j - \theta^\top x_j\right)^2 + \lambda\|\theta\|_2^2 - g(\theta) \\ &= \sum_j w_j\left(y_j - \theta^\top x_j\right)^2 - w_i\left(y_i - \theta^\top x_i\right)^2 + w_i\left(y_i' - \theta^\top x_i'\right)^2 + \lambda\|\theta\|_2^2 \\ &= \sum_{j\neq i} w_j\left(y_j - \theta^\top x_j\right)^2 + w_i\left(y_i' - \theta^\top x_i'\right)^2 + \lambda\|\theta\|_2^2.\end{aligned}$$

is the loss on database D'. Further, both objectives are 2λ-strongly convex due to ℓ_2-norm penalty term. Now, we have that $\nabla g(\theta) = 2w_i\left((\theta^\top x_i - y_i)x_i - (\theta^\top x_i' - y_i')x_i'\right) = 2w_i\left(\theta^\top x_i x_i - \theta^\top x_i' x_i' - y_i x_i + y_i' x_i'\right)$. Since $|y| \leq 1$ per our model, we have

$$\|\nabla g(\theta)\|_2 \leq 2w_i(|\theta^\top x_i'|\|x_i'\| + |\theta^\top x_i|\|x_i\| + 2\sqrt{d}).$$

By the preliminaries section of this proof, we have that $\bar{\theta}(D)$, the minimizer of $G(\theta) + g(\theta)$, has norm at most $B(\lambda)$. On top of this, per our model, we have $\|x\| \leq \sqrt{d}$ (since $x \in [0,1]^d$). Therefore, $|\bar{\theta}(D)^\top x_i|, |\bar{\theta}(D')^\top x_i'| \leq \sqrt{d}B(\lambda)$ by Cauchy–Schwarz, and we have:

$$\left\|\nabla g(\bar{\theta})\right\|_2 \leq 2w_i \left(2dB(\lambda) + 2\sqrt{d}\right). \tag{7}$$

From lemma C.3, we have that

$$\Delta_i \bar{\theta} \leq \frac{1}{2\lambda}\left\|\nabla g(\bar{\theta})\right\|_2.$$

which becomes $\Delta_i \bar{\theta} \leq \frac{2w_i\sqrt{d}}{\lambda}\left(\sqrt{d}B(\lambda) + 1\right)$. □

Implications for privacy

Lemma C.5. *Algorithms 1 is ε_i-differentially private for agent i for all i with*

$$\varepsilon_i = \frac{2w_i\eta\sqrt{d}}{\lambda}(\sqrt{d}B(\lambda) + 1).$$

Proof. Let D, D' be two datasets differing only in agent i's data. Let us call out mechanism M. Let $L(\theta, D)$ be the Ridge regression loss on database D evaluated at θ, and let $\bar{\theta}(D) = \arg\min_\theta L(\theta, D)$. For any given outcome o, we have that

$$\frac{P\left[M(D) = o\right]}{P\left[M(D') = o\right]} = \frac{P\left[\bar{\theta}(D) + Z = o\right]}{P\left[\bar{\theta}(D') + Z = o\right]} = \frac{P\left[Z = o - \bar{\theta}(D)\right]}{P\left[Z = o - \bar{\theta}(D')\right]}.$$

Noting that the probability density function is proportional to $f(z) \propto \exp\left(-\eta\|z\|_2\right)$, this can be written as

$$\begin{aligned}\frac{P\left[M(D) = o\right]}{P\left[M(D') = o\right]} &= \exp\left(-\eta\left\|o - \bar{\theta}(D)\right\| + \eta\left\|o - \bar{\theta}(D')\right\|_2\right) \\ &\leq \exp\left(\eta\left\|\bar{\theta}(D') - \bar{\theta}(D)\right\|_2\right) \\ &\leq \exp\left(\eta \cdot \Delta_i\bar{\theta}\right).\end{aligned}$$

where the second-to-last step comes from the triangle inequality and the last step comes from the definition of ℓ_2-sensitivity with respect to agent i. □

We can now conclude the proof. Picking $w_i = \frac{\varepsilon_i}{\sum_{j=1}^n \varepsilon_j}$ and $\eta = \frac{\lambda}{2\sqrt{d}(\sqrt{d}B(\lambda)+1)}\sum_{j=1}^n \varepsilon_j$, we get the result. Indeed:

- The weights are positive and immediately satisfy, $\sum_i w_i = 1$, as required per our algorithm.
- The level of privacy obtained by agent i is

$$\frac{2w_i\eta}{\lambda}\sqrt{d}(\sqrt{d}B(\lambda)+1) = \frac{2}{\lambda}\frac{\varepsilon_i}{\sum_{j=1}^n \varepsilon_j}\cdot\frac{\lambda}{2\sqrt{d}(\sqrt{d}B(\lambda)+1)}\left(\sum_{j=1}^n \varepsilon_j\right)\cdot\sqrt{d}(\sqrt{d}B(\lambda)+1) = \varepsilon_i.$$

C.2 Proof of Theorem 2.5

We start by deriving a closed-form expression for $\bar{\theta}$. Note that for Algorithm 1, $\bar{\theta}$ is simply the unique solution to the unconstrained minimization problem

$$\arg\min_{\theta\in\mathbb{R}^d} \sum_{i=1}^n w_i\left(y_i - \theta^\top x_i\right)^2 + \lambda\|\theta\|_2^2.$$

Therefore,

$$\bar{\theta} = \arg\min_{\theta\in\mathbb{R}^d} \sum_{i=1}^n w_i\left(y_i - \theta^\top x_i\right)^2 + \lambda\|\theta\|_2^2.$$

Taking the first order-condition, we note that we must have

$$-2\sum_{i=1}^{n} w_i x_i \left(y_i - x_i^\top \bar{\theta}\right) + 2\lambda\theta = 0.$$

This can be rewritten as

$$2\left(\sum_{i=1}^{n} w_i x_i x_i^\top + \lambda I\right)\bar{\theta} = 2\sum_{i=1}^{n} w_i x_i y_i.$$

Because $\lambda > 0$ and $\sum_{i=1}^{n} w_i x_i x_i^\top$ is positive semi-definite, $\sum_{i=1}^{n} w_i x_i x_i^\top + \lambda I$ is invertible, and we obtain the following closed-form expression for $\bar{\theta}$:

$$\bar{\theta} = \left(\sum_{i=1}^{n} w_i x_i x_i^\top + \lambda I\right)^{-1} \sum_{i=1}^{n} w_i x_i y_i.$$

We now rewrite this closed-form expression as a function of θ^*, leveraging the fact that $y_i = x_i^\top \theta^* + Z_i$ for $Z_i \sim N(0, \sigma^2)$ for all observations $i \in [n]$. This yields:

$$\begin{aligned}
\bar{\theta} &= \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \sum_{i=1}^{n} w_i x_i y_i \\
&= \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \sum_{i=1}^{n} w_i x_i (x_i^\top \theta^* + Z_i) \\
&= \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \left(\sum_{i=1}^{n} w_i x_i x_i^\top \theta^* + \sum_{i=1}^{n} w_i x_i Z_i\right) \\
&= \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \left(\left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)\theta^* + \sum_{i=1}^{n} w_i x_i Z_i - \lambda\theta^*\right) \\
&= \theta^* + \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \left(\sum_{i=1}^{n} w_i x_i Z_i - \lambda\theta^*\right).
\end{aligned}$$

In turn, we have that the distance between θ^* and $\hat{\theta}$ satisfies, where Z is the random variable added for privacy as per Algorithms 1:

$$\begin{aligned}
\left\|\hat{\theta} - \theta^*\right\| &= \left\| \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \left(\sum_{i=1}^{n} w_i x_i Z_i - \lambda\theta^*\right) + Z \right\| \\
&\leq \left\| \left(\lambda I + \sum_{i=1}^{n} w_i x_i x_i^\top\right)^{-1} \left(\sum_{i=1}^{n} w_i x_i Z_i - \lambda\theta^*\right) \right\| + \|Z\| \\
&\leq \frac{1}{\lambda + \lambda_{min}\left(\sum_{i=1}^{n} w_i x_i x_i^\top\right)} \left\| \left(\sum_{i=1}^{n} w_i x_i Z_i - \lambda\theta^*\right) \right\| + \|Z\| \\
&\leq \frac{1}{1 + \frac{\lambda_{min}\left(\sum_{i=1}^{n} w_i x_i x_i^\top\right)}{\lambda}} \cdot \|\theta^*\| + \frac{1}{\lambda} \left\| \sum_{i=1}^{n} w_i x_i Z_i \right\| + \|Z\|.
\end{aligned}$$

To conclude the proof, we write concentration inequalities on $\|\sum_{i=1}^{n} w_i x_i Z_i\|$ and on $\|Z\|$:

- For Z: we know per Remark .1 that $\|Z\|_2$ follows a gamma distribution with parameters d and η; therefore, it has mean $\frac{d}{\eta}$ and variance $\frac{d}{\eta^2}$. By Chebyshev, we obtain that with probability at most $\delta/2$, $\|Z\| \geq \frac{d}{\eta} + \sqrt{\frac{2}{\delta}} \cdot \frac{\sqrt{d}}{\eta}$

- For $\|\sum_{i=1}^n w_i x_i Z_i\|$: first note that $S = \sum_{i=1}^n w_i x_i Z_i$ is a multivariate Gaussian random variable. It has mean 0 since the Z_i's have mean 0, and covariance

$$\Sigma = \sum_{i=1}^n \sum_j w_i w_j Cov(Z_i, Z_j) x_i x_j^\top = \sigma^2 \sum_{i=1}^n w_i^2 x_i x_i^\top,$$

where the equality comes from $Cov(Z_i, Z_i) = \sigma^2$ and $Cov(Z_i, Z_j) = 0$ for $i \neq j$ by independence. Note that then since $S(k)$ has mean 0 for all $k \in [d]$,

$$E\left[\|S\|^2\right] = \sum_{k=1}^d E\left[S(k)^2\right] = \sum_{k=1}^d Cov(S(k), S(k)) = \sum_{k=1}^d \Sigma_{kk} = Tr(\Sigma).$$

Now, by Markov's inequality, we have that $P\left[\|S\|^2 \geq \frac{2Tr(\Sigma)}{\delta}\right] \leq \delta/2$, or equivalently $P\left[\|S\| \geq \sqrt{\frac{2Tr(\Sigma)}{\delta}}\right] \leq \delta/2$. To conclude the proof, we simply need to compute the trace of covariance matrix Σ. We have that

$$Tr(\Sigma) = \sigma^2 \sum_{i=1}^n w_i^2 Tr(x_i x_i^\top) = \sigma^2 \sum_{i=1}^n w_i^2 \sum_{k=1}^d x_{ik}^2 = \sigma^2 \sum_{i=1}^n w_i^2 \|x_i\|^2 \leq d\sigma^2 \sum_{i=1}^n w_i^2,$$

using that $\|x\| \leq \sqrt{d}$. This directly implies that

$$P\left[\|S\| \geq \sqrt{\frac{2d\sigma^2\|\vec{w}\|^2}{\delta}}\right] \leq P\left[\|S\| \geq \sqrt{\frac{2Tr(\Sigma)}{\delta}}\right] \leq \delta/2.$$

Therefore, we have that $\frac{1}{\lambda}\|\sum_{i=1}^n w_i x_i Z_i\| \geq \frac{\sigma}{\lambda}\sqrt{\frac{2d}{\delta}}\|\vec{w}\|$ with probability at most $\delta/2$.

The result follows by union bound over the randomness of both Z and $\sum_{i=1}^n w_i x_i Z_i$.

D Additional Experiments

D.1 Comparison to non-personalized differential privacy

In Tables 3 and 4 (respectively on our synthetic and real dataset) where we change the privacy level ε_c for the conservative users. We observe that using personalized privacy still performs significantly better, but the performance improvements diminish as ε_c increases. This is not surprising: as ε_c increases, there is less and less variability across users' privacy levels, leading to less of a need for personalized privacy. Non-personalized privacy estimators start working better as the amount of noise they must add, which scales as a function of ε_c, starts largely decreasing.

In Tables 5 and 6, we change the privacy level ε_m for the pragmatist (or medium privacy) users. We observe once again that using personalized privacy performs significantly better. The performance improvements are consistent across the board, noting that there is still a need for personalized privacy as we still have significant variability across user privacy preferences, with 34 percent of users requiring a stringent privacy level of 0.01 and 23 percent a privacy level of 1.

In Tables 7 and 8, we change the fraction of conservative users f_c. We note that even then, our personalized privacy framework still yields consistent and significant improvements over non-personalized privacy. This is because the existence of users with $\varepsilon_c = 0.01$ forces the non-personalized privacy estimate to still add noise that scales with this most stringent privacy requirement.

Finally, in Tables 9 and 10, we fix a single value of the regularization parameter λ and of the distribution of privacy requirements of the users and show how our results evolve as the number of samples we feed our algorithm increases. Unsurprisingly, the more samples we have access to, the better the performance of both our personalized privacy approach as well as the non-personalized baseline. Our approach continues to see significant and consistent improvements compared to the non-personalized baseline.

Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
1.00	$\mathbf{8.54 \times 10^{2}}$	4.60×10^{5}	$\mathbf{3.32 \times 10^{3}}$	1.77×10^{6}
3.00	$\mathbf{3.88 \times 10^{1}}$	2.08×10^{4}	$\mathbf{3.82 \times 10^{2}}$	2.03×10^{5}
5.00	$\mathbf{9.78}$	5.20×10^{3}	$\mathbf{1.50 \times 10^{2}}$	8.08×10^{4}
7.00	$\mathbf{3.81}$	2.06×10^{3}	$\mathbf{8.36 \times 10^{1}}$	4.46×10^{4}
10.00	$\mathbf{1.49}$	8.18×10^{2}	$\mathbf{4.58 \times 10^{1}}$	2.45×10^{4}
15.00	$\mathbf{5.30 \times 10^{-1}}$	2.75×10^{2}	$\mathbf{2.34 \times 10^{1}}$	1.26×10^{4}
20.00	$\mathbf{2.54 \times 10^{-1}}$	1.29×10^{2}	$\mathbf{1.49 \times 10^{1}}$	8.01×10^{3}
25.00	$\mathbf{1.44 \times 10^{-1}}$	7.45×10^{1}	$\mathbf{1.07 \times 10^{1}}$	5.63×10^{3}
50.00	$\mathbf{2.28 \times 10^{-2}}$	1.11×10^{1}	$\mathbf{3.06}$	1.64×10^{3}
75.00	$\mathbf{9.35 \times 10^{-3}}$	3.78	$\mathbf{1.56}$	8.29×10^{2}
100.00	$\mathbf{5.82 \times 10^{-3}}$	1.80	$\mathbf{1.01}$	5.36×10^{2}
125.00	$\mathbf{4.39 \times 10^{-3}}$	1.06	$\mathbf{7.37 \times 10^{-1}}$	3.91×10^{2}
150.00	$\mathbf{3.68 \times 10^{-3}}$	6.79×10^{-1}	$\mathbf{5.73 \times 10^{-1}}$	3.05×10^{2}
175.00	$\mathbf{3.33 \times 10^{-3}}$	4.76×10^{-1}	$\mathbf{4.67 \times 10^{-1}}$	2.49×10^{2}
200.00	$\mathbf{3.09 \times 10^{-3}}$	3.64×10^{-1}	$\mathbf{3.97 \times 10^{-1}}$	2.10×10^{2}

Table 1: Loss of PDP-OP compared to standard DP on the synthetic dataset with $d = 30, n = 100$, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.

Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.01	$\mathbf{1.19 \times 10^{5}}$	2.18×10^{8}	$\mathbf{1.22 \times 10^{5}}$	2.31×10^{8}
0.05	$\mathbf{1.03 \times 10^{3}}$	1.89×10^{6}	$\mathbf{1.16 \times 10^{3}}$	2.12×10^{6}
0.10	$\mathbf{1.34 \times 10^{2}}$	2.49×10^{5}	$\mathbf{1.70 \times 10^{2}}$	3.08×10^{5}
0.50	$\mathbf{1.30}$	2.40×10^{3}	$\mathbf{3.03}$	5.52×10^{3}
0.60	$\mathbf{8.10 \times 10^{-1}}$	1.47×10^{3}	$\mathbf{2.02}$	3.65×10^{3}
0.70	$\mathbf{5.29 \times 10^{-1}}$	9.26×10^{2}	$\mathbf{1.47}$	2.61×10^{3}
0.80	$\mathbf{3.78 \times 10^{-1}}$	6.36×10^{2}	$\mathbf{1.11}$	1.98×10^{3}
0.90	$\mathbf{2.78 \times 10^{-1}}$	4.59×10^{2}	$\mathbf{8.79 \times 10^{-1}}$	1.54×10^{3}
1.00	$\mathbf{2.15 \times 10^{-1}}$	3.45×10^{2}	$\mathbf{7.12 \times 10^{-1}}$	1.24×10^{3}
2.00	$\mathbf{6.80 \times 10^{-2}}$	5.18×10^{1}	$\mathbf{2.26 \times 10^{-1}}$	3.19×10^{2}
3.00	$\mathbf{5.52 \times 10^{-2}}$	1.73×10^{1}	$\mathbf{1.39 \times 10^{-1}}$	1.52×10^{2}
5.00	$\mathbf{5.54 \times 10^{-2}}$	4.49	$\mathbf{9.65 \times 10^{-2}}$	6.25×10^{1}

Table 2: Loss of PDP-OP compared to standard DP on the Medical cost dataset, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.

Privacy level of Conservatives (ε_c)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.01	$\mathbf{1.44 \times 10^{-2}}$	3.38	$\mathbf{1.15}$	1.05×10^{3}
0.05	$\mathbf{1.41 \times 10^{-2}}$	2.09×10^{-1}	$\mathbf{1.05}$	6.08×10^{1}
0.10	$\mathbf{1.39 \times 10^{-2}}$	4.76×10^{-2}	$\mathbf{1.00}$	1.16×10^{1}
0.20	$\mathbf{1.41 \times 10^{-2}}$	3.40×10^{-2}	$\mathbf{1.03}$	7.21
0.30	$\mathbf{1.34 \times 10^{-2}}$	2.08×10^{-2}	$\mathbf{8.74 \times 10^{-1}}$	1.20
0.40	$\mathbf{1.32 \times 10^{-2}}$	1.65×10^{-2}	$\mathbf{8.29 \times 10^{-1}}$	1.84
0.50	$\mathbf{1.29 \times 10^{-2}}$	1.45×10^{-2}	$\mathbf{7.26 \times 10^{-1}}$	1.20

Table 3: Lower loss compared to standard DP on the synthetic dataset, while varying ε_c (privacy level of the conservative users), keeping $f_c = 0.54, f_m = 0.37, \varepsilon_m = 0.5$ (same parameters as shown in Jorgensen et al. (2015)) and $\lambda = 100$.

D.2 Comparison to Jorgensen et al. (2015)

We provide additional experimental results that further the comparison of our PDP-OP algorithm with that of Jorgensen et al. (2015).

Privacy level of Conservatives (ε_c)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.01	$\mathbf{2.27 \times 10^{-1}}$	2.76×10^{2}	$\mathbf{7.43 \times 10^{-1}}$	9.87×10^{2}
0.05	$\mathbf{2.28 \times 10^{-1}}$	2.00×10^{1}	$\mathbf{7.32 \times 10^{-1}}$	7.19×10^{1}
0.10	$\mathbf{2.12 \times 10^{-1}}$	5.05	$\mathbf{6.92 \times 10^{-1}}$	1.80×10^{1}
0.20	$\mathbf{1.97 \times 10^{-1}}$	1.28	$\mathbf{6.42 \times 10^{-1}}$	4.54
0.30	$\mathbf{1.83 \times 10^{-1}}$	5.84×10^{-1}	$\mathbf{5.75 \times 10^{-1}}$	2.06
0.40	$\mathbf{1.66 \times 10^{-1}}$	3.48×10^{-1}	$\mathbf{5.26 \times 10^{-1}}$	1.17
0.50	$\mathbf{1.55 \times 10^{-1}}$	2.33×10^{-1}	$\mathbf{4.90 \times 10^{-1}}$	7.61×10^{-1}

Table 4: Lower loss compared to standard DP on the Medical cost dataset while varying ε_c (privacy level of the conservative users), keeping $f_c = 0.54, f_m = 0.37, \varepsilon_m = 0.5$ (same parameters as shown in Jorgensen et al. (2015)) and $\lambda = 1$(analogous to Table 3)

Privacy level of pragmatists (ε_m)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.05	$\mathbf{2.11 \times 10^{-2}}$	9.68	**3.23**	2.95×10^{3}
0.10	$\mathbf{1.88 \times 10^{-2}}$	5.76	**2.55**	1.76×10^{3}
0.15	$\mathbf{1.89 \times 10^{-2}}$	8.46	**2.56**	2.60×10^{3}
0.20	$\mathbf{1.79 \times 10^{-2}}$	4.84	**2.25**	1.49×10^{3}
0.25	$\mathbf{1.66 \times 10^{-2}}$	3.17	**1.83**	9.78×10^{2}
0.30	$\mathbf{1.57 \times 10^{-2}}$	1.48	**1.53**	4.52×10^{2}
0.35	$\mathbf{1.58 \times 10^{-2}}$	6.45×10^{-1}	**1.61**	1.97×10^{2}
0.40	$\mathbf{1.48 \times 10^{-2}}$	2.44	**1.33**	7.56×10^{2}
0.45	$\mathbf{1.47 \times 10^{-2}}$	3.21	**1.25**	9.86×10^{2}
0.50	$\mathbf{1.44 \times 10^{-2}}$	3.38	**1.15**	1.05×10^{3}

Table 5: Lower loss compared to standard DP on the synthetic dataset, while varying ε_m (privacy level of the pragmatists), keeping $f_c = 0.54, f_m = 0.37, \varepsilon_c = 0.01$ (same parameters as shown in Jorgensen et al. (2015)) and $\lambda = 100$.

Privacy level of Conservatives (ε_m)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.05	$\mathbf{5.66 \times 10^{-1}}$	4.99×10^{2}	**1.94**	1.75×10^{3}
0.10	$\mathbf{5.18 \times 10^{-1}}$	4.99×10^{2}	**1.76**	1.78×10^{3}
0.15	$\mathbf{4.50 \times 10^{-1}}$	5.02×10^{2}	**1.55**	1.76×10^{3}
0.20	$\mathbf{4.00 \times 10^{-1}}$	4.81×10^{2}	**1.38**	1.71×10^{3}
0.25	$\mathbf{3.60 \times 10^{-1}}$	4.80×10^{2}	**1.22**	1.70×10^{3}
0.30	$\mathbf{3.27 \times 10^{-1}}$	3.94×10^{2}	**1.10**	1.43×10^{3}
0.35	$\mathbf{2.85 \times 10^{-1}}$	4.82×10^{2}	$\mathbf{9.75 \times 10^{-1}}$	1.75×10^{3}
0.40	$\mathbf{2.69 \times 10^{-1}}$	4.83×10^{2}	$\mathbf{9.05 \times 10^{-1}}$	1.71×10^{3}
0.45	$\mathbf{2.53 \times 10^{-1}}$	4.36×10^{2}	$\mathbf{8.28 \times 10^{-1}}$	1.61×10^{3}
0.50	$\mathbf{2.27 \times 10^{-1}}$	2.76×10^{2}	$\mathbf{7.43 \times 10^{-1}}$	9.87×10^{2}

Table 6: Lower loss compared to standard DP on the Medical cost dataset, while varying ε_m (privacy level of the pragmatists), keeping $f_c = 0.54, f_m = 0.37, \varepsilon_c = 0.01$ (same parameters as shown in Jorgensen et al. (2015)) and $\lambda = 1$. (analogous to Table 5)

Fraction of Conservative Users (f_c)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.1	$\mathbf{9.02 \times 10^{-3}}$	7.87×10^{-1}	$\mathbf{4.91 \times 10^{-1}}$	2.43×10^{2}
0.2	$\mathbf{9.41 \times 10^{-3}}$	3.80×10^{-1}	$\mathbf{6.51 \times 10^{-1}}$	1.14×10^{2}
0.3	$\mathbf{1.02 \times 10^{-2}}$	4.56	$\mathbf{8.76 \times 10^{-1}}$	1.41×10^{3}
0.4	$\mathbf{1.14 \times 10^{-2}}$	3.82	**1.27**	1.17×10^{3}
0.5	$\mathbf{1.30 \times 10^{-2}}$	7.00	**1.75**	2.14×10^{3}
0.6	$\mathbf{1.69 \times 10^{-2}}$	9.59	**2.92**	2.90×10^{3}

Table 7: Lower loss compared to standard DP on the synthetic dataset, while varying f_c (fraction of conservative users) for $f_m = 0.37,\ f_l = 1 - f_c - f_m,\ \varepsilon_c = 0.01,\ \varepsilon_m = 0.2, \varepsilon_l = 1.0, \lambda = 100$

Fraction of Conservative Users (f_c)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.1	$\mathbf{1.17 \times 10^{-1}}$	4.95×10^2	$\mathbf{3.47 \times 10^{-1}}$	1.77×10^3
0.2	$\mathbf{1.45 \times 10^{-1}}$	4.89×10^2	$\mathbf{4.46 \times 10^{-1}}$	1.74×10^3
0.3	$\mathbf{1.82 \times 10^{-1}}$	5.00×10^2	$\mathbf{5.78 \times 10^{-1}}$	1.79×10^3
0.4	$\mathbf{2.35 \times 10^{-1}}$	4.80×10^2	$\mathbf{7.71 \times 10^{-1}}$	1.73×10^3
0.5	$\mathbf{3.45 \times 10^{-1}}$	4.93×10^2	**1.16**	1.80×10^3
0.6	$\mathbf{5.61 \times 10^{-1}}$	4.19×10^2	**1.92**	1.51×10^3

Table 8: Lower loss compared to standard DP on the Medical cost dataset, while varying f_c(fraction of conservative users) $f_m = 0.37$, $f_l = 1 - f_c - f_m$, $\varepsilon_c = 0.01$, $\varepsilon_m = 0.2$, $\varepsilon_l = 1.0$, $\lambda = 1$.

Fraction of training samples (n)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.1	$\mathbf{4.28 \times 10^{-1}}$	1.22×10^1	$\mathbf{1.32 \times 10^2}$	3.73×10^3
0.2	$\mathbf{1.41 \times 10^{-1}}$	1.15×10^2	$\mathbf{4.02 \times 10^1}$	3.45×10^4
0.3	$\mathbf{5.01 \times 10^{-2}}$	8.97	$\mathbf{1.46 \times 10^1}$	2.89×10^3
0.4	$\mathbf{4.64 \times 10^{-2}}$	2.37	$\mathbf{1.27 \times 10^1}$	7.24×10^2
0.5	$\mathbf{3.49 \times 10^{-2}}$	2.14×10^1	**8.11**	6.61×10^3
0.6	$\mathbf{2.12 \times 10^{-2}}$	1.59×10^1	**5.16**	5.06×10^3
0.7	$\mathbf{2.00 \times 10^{-2}}$	7.97	**4.66**	2.52×10^3
0.8	$\mathbf{1.64 \times 10^{-2}}$	1.03×10^1	**3.36**	2.95×10^3
0.9	$\mathbf{1.10 \times 10^{-2}}$	6.36	**2.52**	1.93×10^3
1.0	$\mathbf{1.01 \times 10^{-2}}$	1.98	**2.16**	6.17×10^2

Table 9: Lower loss compared to standard DP on the synthetic dataset, while varying the fraction of the training set samples we use. For example, here $n = 0.3$ means that we use a 0.3 fraction of the training set. We fix $f_c = 0.34$, $f_m = 0.43$, $f_l = 0.23$, $\varepsilon_c = 0.01$, $\varepsilon_m = 0.2$, $\varepsilon_l = 1.0$ and $\lambda = 100$.

Fraction of training samples (n)	Unregularized test loss (PDP-OP)	Unregularized test loss (non-personalized)	Regularized test loss (PDP-OP)	Regularized test loss (non-personalized)
0.1	$\mathbf{1.83 \times 10^1}$	1.45×10^4	$\mathbf{6.64 \times 10^1}$	5.21×10^4
0.2	**4.63**	6.62×10^3	$\mathbf{1.65 \times 10^1}$	2.37×10^4
0.3	**2.05**	5.23×10^3	**7.29**	1.85×10^4
0.4	**1.21**	1.60×10^3	**4.28**	5.77×10^3
0.5	$\mathbf{7.85 \times 10^{-1}}$	1.96×10^3	**2.75**	6.98×10^3
0.6	$\mathbf{5.40 \times 10^{-1}}$	1.36×10^3	**1.84**	4.91×10^3
0.7	$\mathbf{4.02 \times 10^{-1}}$	9.64×10^2	**1.38**	3.47×10^3
0.8	$\mathbf{3.16 \times 10^{-1}}$	7.26×10^2	**1.06**	2.60×10^3
0.9	$\mathbf{2.53 \times 10^{-1}}$	6.15×10^2	$\mathbf{8.56 \times 10^{-1}}$	2.24×10^3
1.0	$\mathbf{2.14 \times 10^{-1}}$	3.82×10^2	$\mathbf{6.99 \times 10^{-1}}$	1.36×10^3

Table 10: Lower loss compared to standard DP on the Medical cost dataset, while varying the number of samples we use. For example, here $n = 0.3$ means that we use a 0.3 fraction of the training set. We fix $f_c = 0.34$, $f_m = 0.43$, $f_l = 0.23$, $\varepsilon_c = 0.01$, $\varepsilon_m = 0.2$, $\varepsilon_l = 1.0$ and $\lambda = 1$.

Improvements in loss. In Figure 2 and Figure 3 we vary the privacy level ε_c for the conservative users, and observe that our algorithm always leads to a lower loss compared to Jorgensen et al. (2015). We further note that the performance improvements diminish as ε_c increases, as there is less variability in the users' privacy levels and we get closer to the non-personalized case.

Figures 4 and 5 show similar insights when varying the parameter ε_m, that controls the privacy level of medium-privacy (or pragmatic) users.

In Figures 6 and 7 we vary the fraction of users with strong privacy requirements. We remark that our relative performance improvement compared to Jorgensen et al. (2015) becomes bigger as a larger fraction of users has high privacy requirements, highlighting the benefit of our framework in stringent privacy regimes.

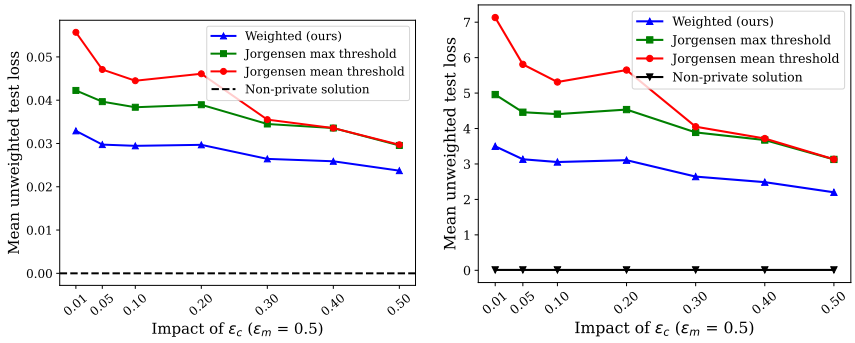
Finally, Figures 8 and 9 vary the fraction of the training set used. We note that our loss improvements compared to Jorgensen et al. (2015) become less noticeable as n increases. This is perhaps

Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (Jorgensen max)	Unregularized test loss (Jorgensen mean)	Unregularized non-private test loss	Regularized test loss (PDP-OP)	Regularized test loss (Jorgensen max)	Regularized test loss (Jorgensen mean)	Regularized non-private test loss
1.00	$\mathbf{8.54 \times 10^2}$	1.09×10^3	1.89×10^3	5.73×10^{-32}	$\mathbf{3.32 \times 10^3}$	4.16×10^3	7.55×10^3	2.01×10^{-3}
3.00	$\mathbf{3.88 \times 10^1}$	4.82×10^1	8.64×10^1	5.73×10^{-32}	$\mathbf{3.82 \times 10^2}$	4.79×10^2	8.64×10^2	2.11×10^{-3}
5.00	**9.78**	1.21×10^1	2.15×10^1	5.73×10^{-32}	$\mathbf{1.50 \times 10^2}$	1.89×10^2	3.40×10^2	2.17×10^{-3}
7.00	**3.81**	4.81	8.85	5.73×10^{-32}	$\mathbf{8.36 \times 10^1}$	1.05×10^2	1.89×10^2	2.20×10^{-3}
10.00	**1.49**	1.88	3.44	5.73×10^{-32}	$\mathbf{4.58 \times 10^1}$	5.75×10^1	1.03×10^2	2.24×10^{-3}
15.00	$\mathbf{5.30 \times 10^{-1}}$	6.61×10^{-1}	1.16	5.73×10^{-32}	$\mathbf{2.34 \times 10^1}$	2.97×10^1	5.33×10^1	2.28×10^{-3}
20.00	$\mathbf{2.54 \times 10^{-1}}$	3.12×10^{-1}	5.64×10^{-1}	5.73×10^{-32}	$\mathbf{1.49 \times 10^1}$	1.87×10^1	3.39×10^1	2.31×10^{-3}
25.00	$\mathbf{1.44 \times 10^{-1}}$	1.81×10^{-2}	3.27×10^{-1}	5.73×10^{-32}	$\mathbf{1.07 \times 10^1}$	1.31×10^1	2.39×10^1	2.33×10^{-3}
50.00	$\mathbf{2.28 \times 10^{-2}}$	2.82×10^{-2}	4.92×10^{-2}	5.73×10^{-32}	**3.06**	3.83	6.95	2.37×10^{-3}
75.00	$\mathbf{9.35 \times 10^{-3}}$	1.12×10^{-2}	1.81×10^{-2}	5.73×10^{-32}	**1.56**	1.95	3.56	2.39×10^{-3}
100.00	$\mathbf{5.82 \times 10^{-3}}$	6.61×10^{-3}	1.01×10^{-2}	5.73×10^{-32}	**1.01**	1.26	2.28	2.40×10^{-3}
200.00	$\mathbf{3.09 \times 10^{-3}}$	3.25×10^{-3}	3.92×10^{-3}	5.73×10^{-32}	$\mathbf{3.97 \times 10^{-1}}$	4.97×10^{-1}	9.02×10^{-1}	2.42×10^{-3}
300.00	$\mathbf{2.68 \times 10^{-3}}$	2.76×10^{-3}	3.04×10^{-3}	5.73×10^{-32}	$\mathbf{2.42 \times 10^{-1}}$	3.02×10^{-1}	5.46×10^{-1}	2.42×10^{-3}
400.00	$\mathbf{2.56 \times 10^{-3}}$	2.60×10^{-3}	2.76×10^{-3}	5.73×10^{-32}	$\mathbf{1.75 \times 10^{-1}}$	2.16×10^{-1}	3.92×10^{-1}	2.42×10^{-3}
500.00	$\mathbf{2.51 \times 10^{-3}}$	2.54×10^{-3}	2.62×10^{-3}	5.73×10^{-32}	$\mathbf{1.36 \times 10^{-1}}$	1.70×10^{-1}	3.06×10^{-1}	2.43×10^{-3}

Table 11: Lower loss compared to Jorgensen et al. (2015) on the synthetic dataset with $d = 30,\ n = 100$, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.

Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (Jorgensen max)	Unregularized test loss (Jorgensen mean)	Unregularized non-private test loss	Regularized test loss (PDP-OP)	Regularized test loss (Jorgensen max)	Regularized test loss (Jorgensen mean)	Regularized non-private test loss
0.01	$\mathbf{1.19 \times 10^5}$	1.50×10^5	2.87×10^5	9.43×10^{-3}	$\mathbf{1.22 \times 10^5}$	1.52×10^5	2.93×10^5	1.08×10^{-2}
0.05	$\mathbf{1.03 \times 10^3}$	1.26×10^3	2.43×10^3	9.43×10^{-3}	$\mathbf{1.16 \times 10^3}$	1.42×10^3	2.75×10^3	1.51×10^{-2}
0.10	$\mathbf{1.34 \times 10^2}$	1.66×10^2	3.14×10^2	9.43×10^{-3}	$\mathbf{1.70 \times 10^2}$	2.11×10^2	4.00×10^2	1.90×10^{-2}
0.50	**1.30**	1.67	3.10	9.43×10^{-3}	**3.03**	3.80	7.19	3.46×10^{-2}
0.60	$\mathbf{8.10 \times 10^{-1}}$	9.89×10^{-1}	1.85	9.43×10^{-3}	**2.02**	2.50	4.76	3.67×10^{-2}
0.70	$\mathbf{5.29 \times 10^{-1}}$	6.55×10^{-1}	1.23	9.43×10^{-3}	**1.47**	1.80	3.43	3.86×10^{-2}
0.80	$\mathbf{3.78 \times 10^{-1}}$	4.65×10^{-1}	8.71×10^{-1}	9.43×10^{-3}	**1.11**	1.39	2.63	4.02×10^{-2}
0.90	$\mathbf{2.78 \times 10^{-1}}$	3.42×10^{-1}	6.30×10^{-1}	9.43×10^{-3}	$\mathbf{8.79 \times 10^{-1}}$	1.08	2.03	4.17×10^{-2}
1.00	$\mathbf{2.15 \times 10^{-1}}$	2.61×10^{-1}	4.76×10^{-1}	9.43×10^{-3}	$\mathbf{7.12 \times 10^{-1}}$	8.70×10^{-1}	1.62	4.30×10^{-2}
2.00	$\mathbf{6.80 \times 10^{-2}}$	7.53×10^{-2}	1.06×10^{-1}	9.43×10^{-3}	$\mathbf{2.26 \times 10^{-1}}$	2.65×10^{-1}	4.59×10^{-1}	5.19×10^{-2}
3.00	$\mathbf{5.52 \times 10^{-2}}$	5.72×10^{-2}	6.87×10^{-2}	9.43×10^{-3}	$\mathbf{1.39 \times 10^{-1}}$	1.59×10^{-1}	2.52×10^{-1}	5.69×10^{-2}
5.00	$\mathbf{5.54 \times 10^{-2}}$	5.61×10^{-2}	5.89×10^{-2}	9.43×10^{-3}	$\mathbf{9.65 \times 10^{-2}}$	1.05×10^{-1}	1.43×10^{-1}	6.27×10^{-2}

Table 12: Lower loss compared to Jorgensen et al. (2015) on the MedicalCost dataset, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.

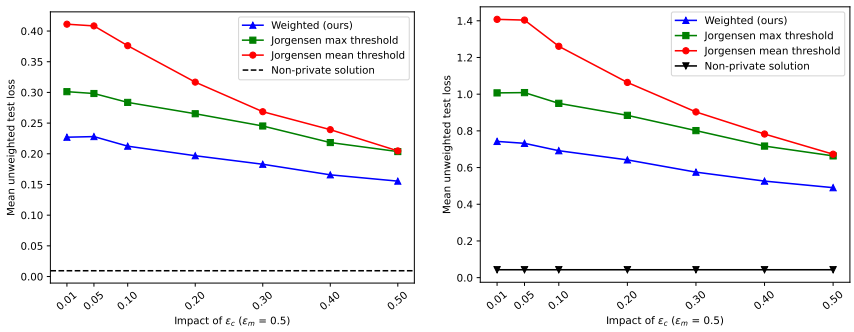


(a) Unregularized Loss ($\lambda = 50$) (b) Regularized Loss ($\lambda = 50$)

Figure 2: Lower loss compared to Jorgensen et al. (2015) on the synthetic dataset while varying ε_c (privacy level of the conservative users) , keeping $\varepsilon_m = 0.5,\ \varepsilon_l = 1.0,\ f_c = 0.54,\ f_m = 0.37, f_l = 0.09$.

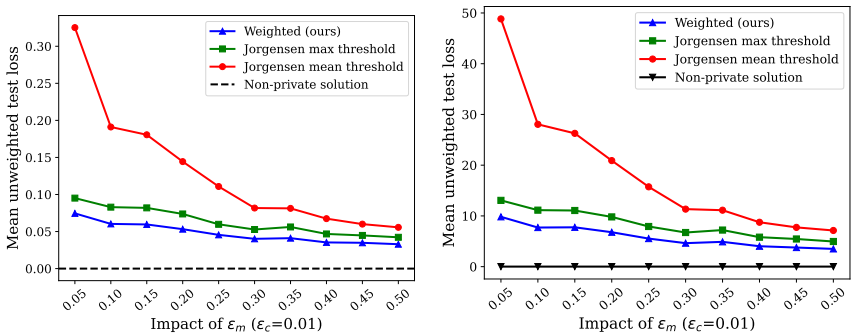
unsurprising, as the more data points we use, the lesser the impact of adding noise for privacy is, and there is much less leeway for improvement across different techniques for privacy.

Improvements in variability. We start with Figures 12 and 13 where we vary the privacy level ε_c for the conservative users. Then, on Figures 14 and 15, we focus on the case of varying ε_m. On Figures 16 and 17, we vary the fraction of users that have high privacy requirements. Finally, on Figures 18 and 19, we vary the fraction of the training set used. On all figures, we note that the standard deviation of the loss of our PDP-OP algorithm is lower than Jorgensen et al. (2015).



(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 3: Lower loss compared to Jorgensen et al. (2015) on the Medical costs dataset while varying ε_c (privacy level of the conservative users), keeping $\varepsilon_m = 0.5, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.

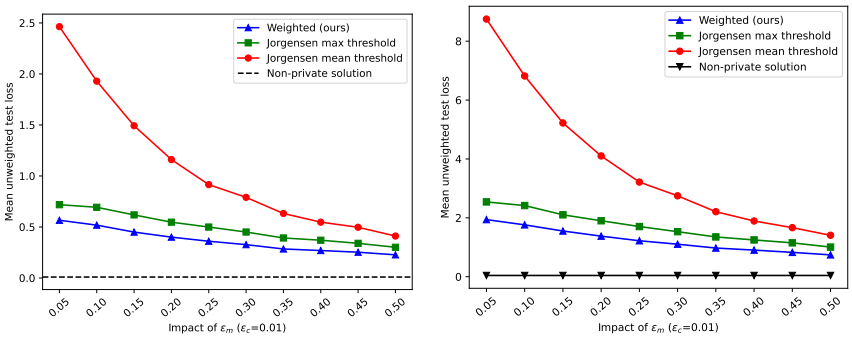


(a) Unregularized Loss ($\lambda = 50$) (b) Regularized Loss ($\lambda = 50$)

Figure 4: Lower loss compared to Jorgensen et al. (2015) on the synthetic dataset while varying ε_m (privacy level of pragmatists), keeping $\varepsilon_c = 0.01, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.

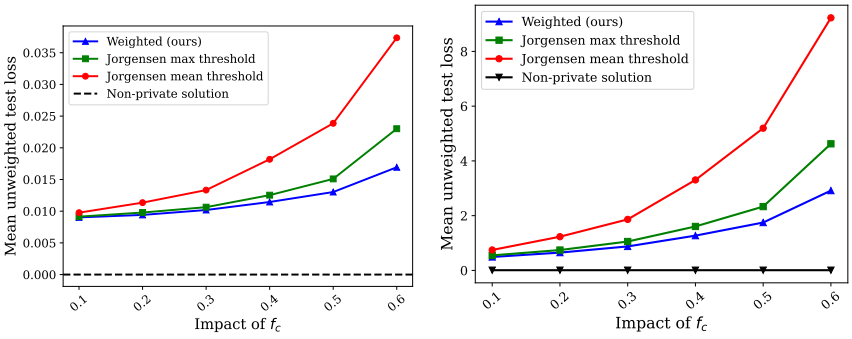
Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (Jorgensen max)	Unregularized test loss (Jorgensen mean)	Regularized test loss (PDP-OP)	Regularized test loss (Jorgensen max)	Regularized test loss (Jorgensen mean)
1.00	$\mathbf{9.69 \times 10^2}$	1.34×10^3	2.19×10^3	$\mathbf{1.53 \times 10^3}$	2.01×10^3	3.47×10^3
3.00	$\mathbf{4.33 \times 10^1}$	5.63×10^1	9.98×10^1	$\mathbf{1.45 \times 10^2}$	1.99×10^2	3.39×10^2
5.00	$\mathbf{1.13 \times 10^1}$	1.41×10^1	2.52×10^1	$\mathbf{5.65 \times 10^1}$	7.69×10^1	1.30×10^2
7.00	**4.43**	5.50	1.02×10^1	$\mathbf{3.15 \times 10^1}$	4.20×10^1	7.14×10^1
10.00	**1.74**	2.18	4.09	$\mathbf{1.72 \times 10^1}$	2.32×10^1	3.98×10^1
15.00	$\mathbf{6.14 \times 10^{-1}}$	7.69×10^{-1}	1.30	**8.73**	1.20×10^1	2.03×10^1
20.00	$\mathbf{2.88 \times 10^{-1}}$	3.66×10^{-1}	6.44×10^{-1}	**5.52**	7.43	1.29×10^1
25.00	$\mathbf{1.59 \times 10^{-1}}$	2.15×10^{-1}	3.79×10^{-1}	**3.91**	5.22	9.15
50.00	$\mathbf{2.37 \times 10^{-2}}$	3.10×10^{-2}	5.50×10^{-2}	**1.13**	1.53	2.62
75.00	$\mathbf{8.54 \times 10^{-3}}$	1.11×10^{-2}	1.88×10^{-2}	$\mathbf{5.73 \times 10^{-1}}$	7.85×10^{-1}	1.34
100.00	$\mathbf{4.70 \times 10^{-3}}$	5.64×10^{-3}	9.85×10^{-3}	$\mathbf{3.72 \times 10^{-1}}$	5.08×10^{-1}	8.75×10^{-1}
200.00	$\mathbf{1.33 \times 10^{-3}}$	1.56×10^{-3}	2.40×10^{-3}	$\mathbf{1.45 \times 10^{-1}}$	1.99×10^{-1}	3.41×10^{-1}
300.00	$\mathbf{7.59 \times 10^{-4}}$	8.62×10^{-4}	1.26×10^{-3}	$\mathbf{8.77 \times 10^{-2}}$	1.21×10^{-1}	2.07×10^{-1}
400.00	$\mathbf{5.41 \times 10^{-4}}$	6.14×10^{-4}	8.72×10^{-4}	$\mathbf{6.38 \times 10^{-2}}$	8.48×10^{-2}	1.48×10^{-1}
500.00	$\mathbf{4.23 \times 10^{-4}}$	4.79×10^{-4}	6.45×10^{-4}	$\mathbf{4.92 \times 10^{-2}}$	6.74×10^{-2}	1.16×10^{-1}

Table 13: Improvements in variability of the test loss: Standard deviation of our algorithm is always lower, results on synthetic dataset with $d = 30,\ n = 100$), while varying the regularization parameter λ, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43$.



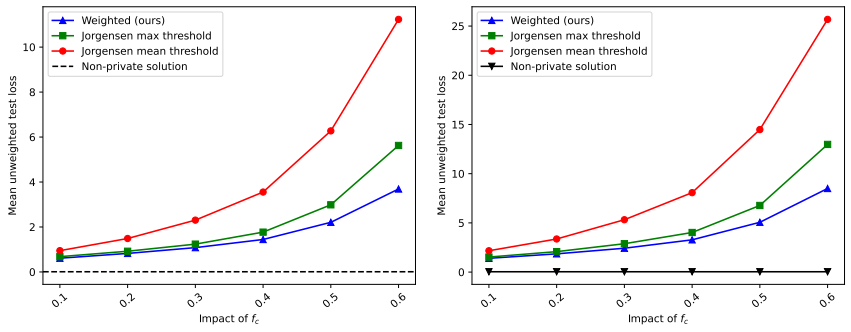
(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 5: Lower loss compared to Jorgensen et al. (2015) for the Medical costs dataset while varying ε_m (privacy level of pragmatists), keeping $\varepsilon_c = 0.01, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.



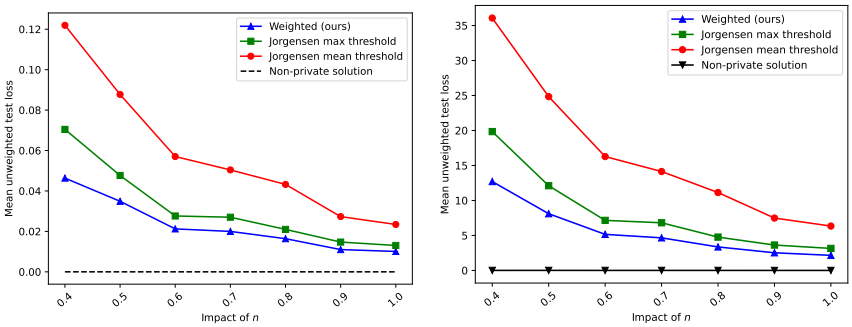
(a) Unregularized Loss ($\lambda = 100$) (b) Regularized Loss ($\lambda = 100$)

Figure 6: Lower loss compared to Jorgensen et al. (2015) on the synthetic dataset while varying f_c(fraction of conservative users), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_m = 0.37, f_l = 1 - f_m - f_c = 0.63 - f_c$.



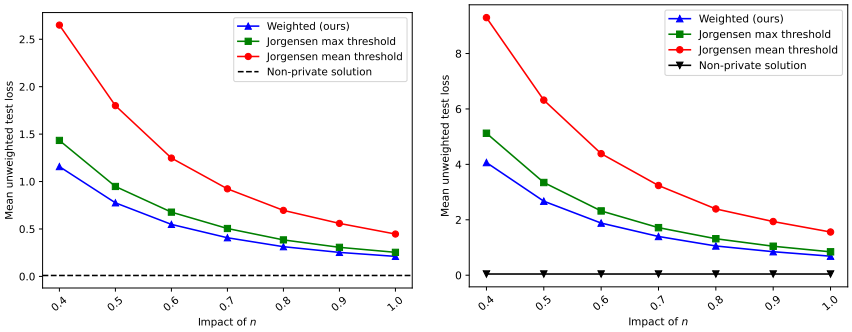
(a) Unregularized Loss ($\lambda = 0.5$) (b) Regularized Loss ($\lambda = 0.5$)

Figure 7: Lower loss compared to Jorgensen et al. (2015) on the Medical costs dataset while f_c (fraction of conservative users), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_m = 0.37, f_l = 1 - f_m - f_c = 0.63 - f_c$.



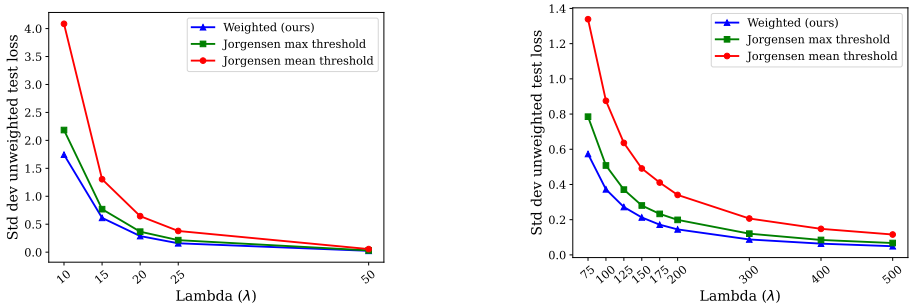
(a) Unregularized Loss ($\lambda = 100$) (b) Regularized Loss ($\lambda = 100$))

Figure 8: Lower loss compared to Jorgensen et al. (2015) on the synthetic dataset while varying the parameter n (the fraction of training samples used), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_c = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.



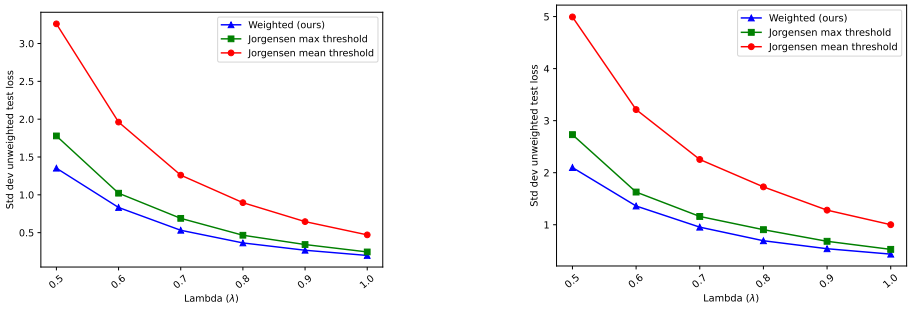
(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$))

Figure 9: Lower loss compared to Jorgensen et al. (2015) on the Medical costs dataset while varying n (the fraction of training samples used), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_c = 1.0, f_c = 0.34, f_m = 0.43$.



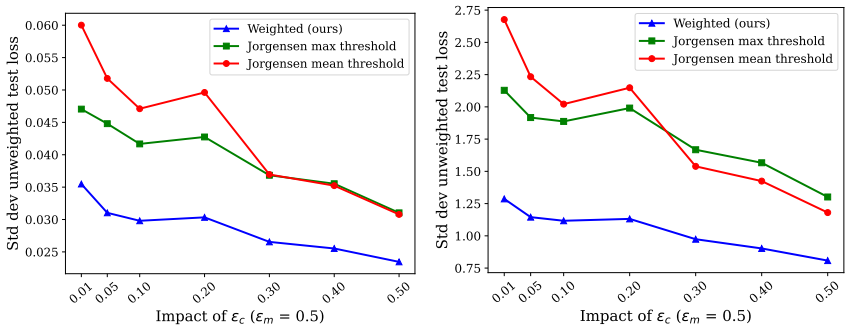
(a) Unregularized Loss ($d = 30, n = 100$) (b) Regularized Loss ($d = 30, n = 100$)

Figure 10: Lower standard deviation compared to Jorgensen et al. (2015) on the synthetic dataset, while varying the regularization parameter λ, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.



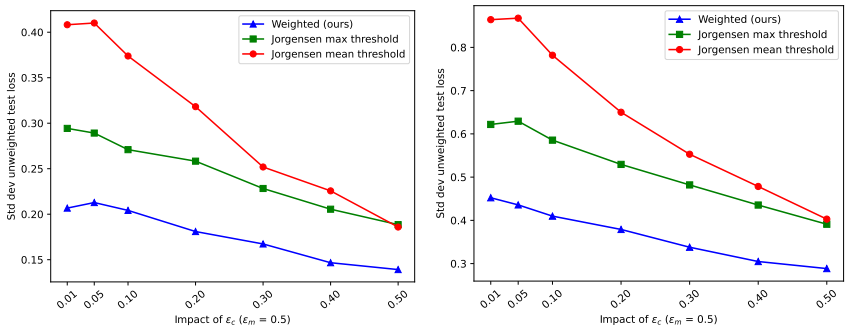
(a) Unregularized Loss (b) Regularized Loss

Figure 11: Lower standard deviation compared to Jorgensen et al. (2015) on the Medical cost dataset, while varying the regularization parameter λ, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.



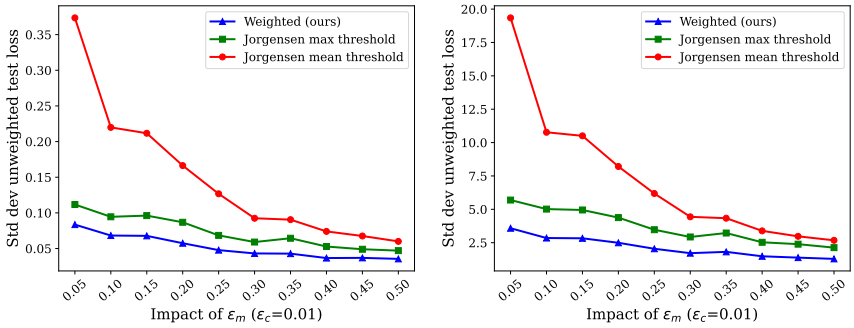
(a) Unregularized Loss ($\lambda = 50$) (b) Regularized Loss ($\lambda = 50$)

Figure 12: Lower standard deviation compared to Jorgensen et al. (2015) on the synthetic dataset, while varying the ε_c (the privacy level of conservative users), keeping $\varepsilon_m = 0.5, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.



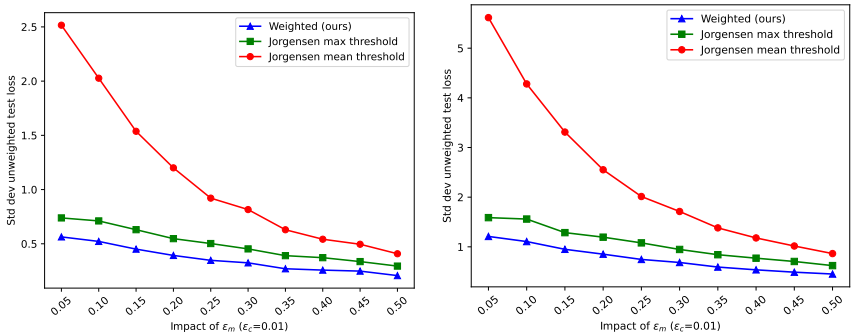
(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 13: Lower standard deviation compared to Jorgensen et al. (2015) on the Medical cost dataset, while varying the ε_c (the privacy level of conservative users), keeping $\varepsilon_m = 0.5, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.



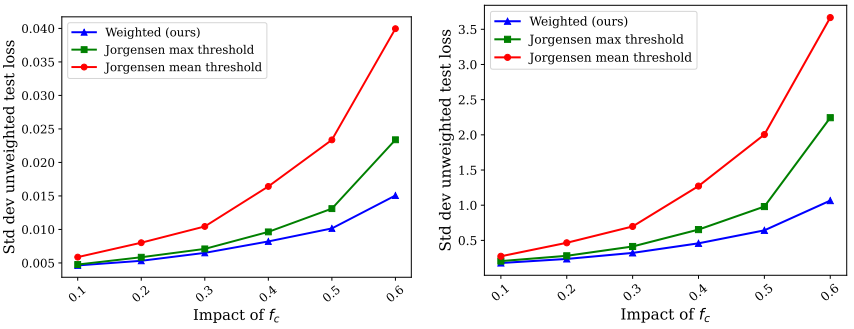
(a) Unregularized Loss ($\lambda = 50$) (b) Regularized Loss ($\lambda = 50$)

Figure 14: Lower standard deviation compared to Jorgensen et al. (2015) on the synthetic dataset, while varying ε_m (privacy level privacy level of the pragmatists), keeping $\varepsilon_c = 0.01, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.



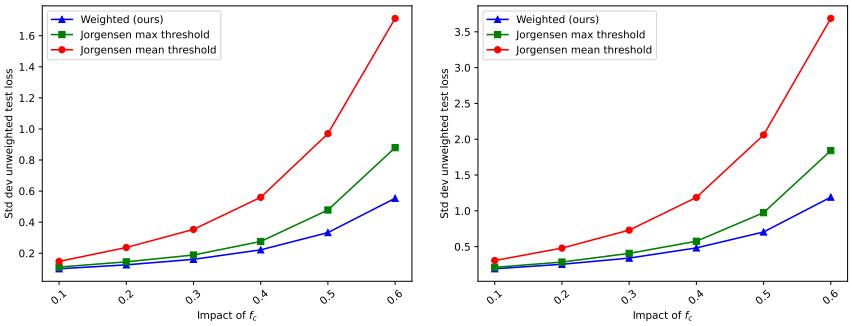
(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 15: Lower standard deviation compared to Jorgensen et al. (2015) on the Medical cost dataset, while varying ε_m (privacy level privacy level of the pragmatists), keeping $\varepsilon_c = 0.01, \varepsilon_l = 1.0, f_c = 0.54, f_m = 0.37, f_l = 0.09$.



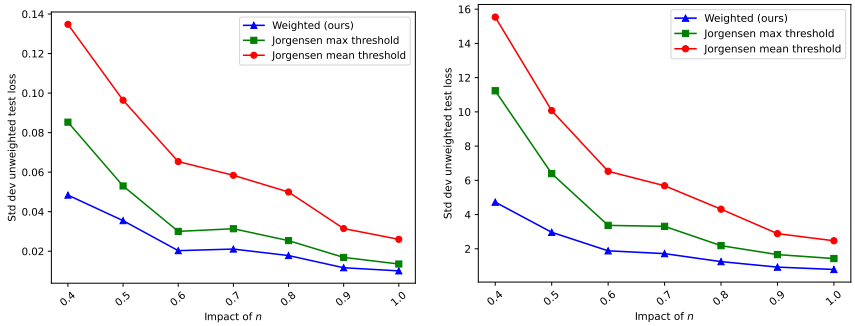
(a) Unregularized Loss ($\lambda = 100$) (b) Regularized Loss ($\lambda = 100$)

Figure 16: Lower standard deviation compared to Jorgensen et al. (2015) on the synthetic dataset, while varying f_c (the fraction of conservative users), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_m = 0.37, f_l = 1 - f_c - f_m = 0.63 - f_c$.



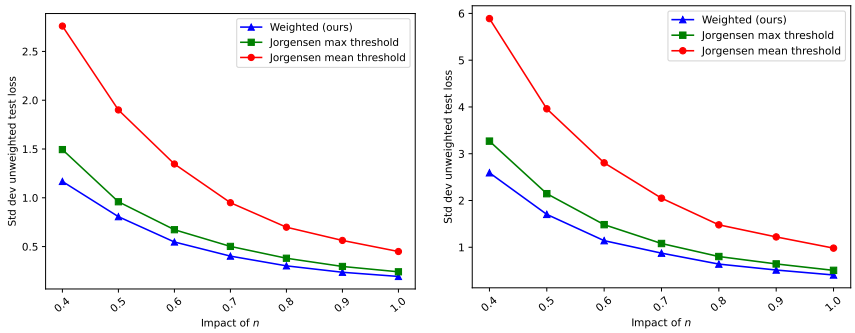
(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 17: Lower standard deviation compared to Jorgensen et al. (2015) on the Medical cost dataset, while varying f_c (the fraction of conservative users), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_m = 0.37, f_l = 1 - f_c - f_m = 0.63 - f_c$.



(a) Unregularized Loss ($\lambda = 100$) (b) Regularized Loss ($\lambda = 100$)

Figure 18: Lower standard deviation compared to Jorgensen et al. (2015) on the synthetic dataset, while varying the parameter n (the fraction of training samples used), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.



(a) Unregularized Loss ($\lambda = 1$) (b) Regularized Loss ($\lambda = 1$)

Figure 19: Lower standard deviation compared to Jorgensen et al. (2015) on the Medical cost dataset, while varying the parameter n (the fraction of training samples used), keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43, f_l = 0.23$.

Regularization parameter Lambda (λ)	Unregularized test loss (PDP-OP)	Unregularized test loss (Jorgensen max)	Unregularized test loss (Jorgensen mean)	Regularized test loss (PDP-OP)	Regularized test loss (Jorgensen max)	Regularized test loss (Jorgensen mean)
0.01	$\mathbf{1.25 \times 10^5}$	1.58×10^5	3.12×10^5	$\mathbf{1.25 \times 10^5}$	1.56×10^5	3.01×10^5
0.05	$\mathbf{1.08 \times 10^3}$	1.32×10^3	2.60×10^3	$\mathbf{1.13 \times 10^3}$	1.37×10^3	2.76×10^3
0.10	$\mathbf{1.38 \times 10^2}$	1.75×10^2	3.28×10^2	$\mathbf{1.54 \times 10^2}$	1.96×10^2	3.75×10^2
0.50	**1.35**	1.78	3.26	**2.10**	2.73	4.99
0.60	$\mathbf{8.33 \times 10^{-1}}$	1.02	1.96	**1.36**	1.63	3.21
0.70	$\mathbf{5.31 \times 10^{-1}}$	6.89×10^{-1}	1.26	$\mathbf{9.54 \times 10^{-1}}$	1.16	2.25
0.80	$\mathbf{3.64 \times 10^{-1}}$	4.66×10^{-1}	8.96×10^{-1}	$\mathbf{6.93 \times 10^{-1}}$	9.05×10^{-1}	1.73
0.90	$\mathbf{2.67 \times 10^{-1}}$	3.43×10^{-1}	6.46×10^{-1}	$\mathbf{5.39 \times 10^{-1}}$	6.82×10^{-1}	1.28
1.00	$\mathbf{1.98 \times 10^{-1}}$	2.45×10^{-1}	4.71×10^{-1}	$\mathbf{4.35 \times 10^{-1}}$	5.27×10^{-1}	1.00
2.00	$\mathbf{3.96 \times 10^{-2}}$	5.00×10^{-2}	8.04×10^{-2}	$\mathbf{1.06 \times 10^{-1}}$	1.28×10^{-1}	2.43×10^{-1}
3.00	$\mathbf{2.11 \times 10^{-2}}$	2.33×10^{-2}	3.84×10^{-2}	$\mathbf{4.80 \times 10^{-2}}$	6.15×10^{-2}	1.17×10^{-1}
5.00	$\mathbf{1.11 \times 10^{-2}}$	1.27×10^{-2}	1.78×10^{-2}	$\mathbf{2.02 \times 10^{-2}}$	2.50×10^{-2}	4.77×10^{-2}

Table 14: Improvements in variability of the test loss: Standard deviation of our algorithm is always lower, results on Medical cost dataset, while varying the regularization parameter λ, keeping $\varepsilon_c = 0.01, \varepsilon_m = 0.2, \varepsilon_l = 1.0, f_c = 0.34, f_m = 0.43$.