Differentially Private Domain Adaptation with Theoretical Guarantees

Raef Bassily [1] **Corinna Cortes** [2] **Anqi Mao** [3] **Mehryar Mohri** [2 3]

Abstract

In many applications, the labeled data at the learner's disposal is subject to privacy constraints and is relatively limited. To derive a more accurate predictor for the target domain, it is often beneficial to leverage publicly available labeled data from an alternative domain, somewhat close to the target domain. This is the modern problem of supervised domain adaptation from a public source to a private target domain. We present two (ε, δ)-differentially private adaptation algorithms for supervised adaptation, for which we make use of a general optimization problem, recently shown to benefit from favorable theoretical learning guarantees. Our first algorithm is designed for regression with linear predictors and shown to solve a convex optimization problem. Our second algorithm is a more general solution for loss functions that may be non-convex but Lipschitz and smooth. While our main objective is a theoretical analysis, we also report the results of several experiments. We first show that the non-private versions of our algorithms match state-of-the-art performance in supervised adaptation and that for larger values of the target sample size or ε, the performance of our private algorithms remains close to that of their non-private counterparts.

1. Introduction

In many applications, the labeled data at hand is not sufficient to train an accurate model for a target domain. Instead, a large amount of labeled data may be available from another domain, a *source domain*, somewhat close to the target domain. The problem then consists of using the labeled data available from both the source and target domains to come up with a more accurate predictor for the target domain. This is the setting of *supervised domain adaptation*.

[1]The Ohio State University [2]Google Research, New York, NY; [3]Courant Institute of Mathematical Sciences, New York, NY. Correspondence to: Anqi Mao <aqmao@cims.nyu.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

The problem faced in practice is often even more challenging, since the labeled data from the target domain can be sensitive and subject to privacy constraints (Bassily, Mohri, and Suresh, 2022). For example, a corporation such as an airline company, or an institution such as a hospital, may seek to train a classifier based on private labeled data it has collected, as well as a large amount of data available from a public domain. To share the classifier internally, let alone share it publicly to the benefit of other institutions or individuals, it may have to train the classifier with privacy guarantees. In the absence of the public domain data and without the adaptation scenario, the framework of differential privacy (Dwork, McSherry, Nissim, and Smith, 2006b; Dwork and Roth, 2014b) can be used to privately learn a classifier that can be shared publicly. But, how can we rigorously design a differentially private algorithm for supervised domain adaptation?

This paper deals precisely with the problem of devising a private (supervised) domain adaptation algorithm with theoretical guarantees for this scenario. Our scenario covers any standard privacy learning scenario where additional data from another public source is sought.

The problem of domain adaptation has been theoretically investigated in a series of publications in the past and the notion of *discrepancy* was shown to be a key divergence measure to the analysis of adaptation (Kifer et al., 2004; Blitzer et al., 2008; Ben-David et al., 2010; Mansour et al., 2009; Cortes & Mohri, 2011; Mohri & Muñoz Medina, 2012; Cortes & Mohri, 2014; Cortes et al., 2019; Zhang et al., 2020b). Building on this prior work, Awasthi, Cortes, and Mohri (2024) recently gave a general theoretical analysis of supervised adaptation that holds for any method relying on reweighting the source and target labeled samples, including reweighting methods that depend on the predictor selected. Thus, the analysis covers a large number of algorithms in adaptation, including importance weighting (Sugiyama et al., 2007b; Lu et al., 2021; Sugiyama et al., 2007a; Cortes et al., 2010; Zhang et al., 2020a), KLIEP (Sugiyama et al., 2007b), Kernel Mean Matching (KMM) (Huang et al., 2006), discrepancy minimization (DM) or generalized discrepancy minimization (Cortes & Mohri, 2014; Cortes et al., 2019). The authors also suggested a general optimization problem that consists of minimizing the right-hand side of their learning bound.

Contributions. We present two (ε, δ)-differentially private adaptation algorithms for supervised adaptation, based on the optimization problem of Awasthi et al. (2024). We first consider a regression setting with linear predictors (Section 4). We show that after a suitable reparameterization of the weights assigned to the sample losses, the optimization problem for adaptation can be formulated as a joint convex optimization problem over the choice of the predictor and that of the reparameterized weights. We then provide an (ε, δ)-differentially private adaptation algorithm, CnvxAdap, using that convex formulation that can be viewed as a variant of noisy projected gradient descent. We note that noisy gradient descent is a general technique that has been well studied in the literature of private optimization (Bassily et al., 2014; Abadi et al., 2016; Wang et al., 2017; Bassily et al., 2019). We prove a formal convergence guarantee for our private algorithm in terms of ε and δ and the sizes of the source and target samples.

We then consider in Section 5 a more general setting where the loss function may be a non-convex function of the parameters and is only assumed to be Lipschitz and smooth. This covers the familiar case where the logistic loss is applied to the output of neural networks, that is cross-entropy with softmax. We show that, remarkably, here too, that reparameterization of the weights combined with the use of the softmax can help us design an (ε, δ)-differentially private algorithm, NCnvxAdap, that benefits from favorable convergence guarantees to stationary points of the objective based on the gradient mapping criterion (Beck, 2017).

While the main objective of our work is a theoretical analysis, we also report extensive empirical evaluations. In Section 6, we present the results of extensive experiments conducted for both our convex and non-convex private algorithms. We first show that the non-private version of our convex algorithm, CnvxAdap ($\epsilon = +\infty$), matches state-of-the-art performance in supervised adaptation and next, that our private algorithm performs comparably to its non-private counterpart, showcasing the effectiveness of our privacy-preserving approach. Similarly, for our non-convex algorithm, NCnvxAdap, the non-private version matches state-of-the-art performance in supervised adaptation and the performance of our private algorithm approaches that of its non-private version as the target sample size and privacy budget (ε) increase.

Related work. Private *density estimation* using a small amount of public data has been studied in several recent publications, in particular for learning Gaussian distributions or mixtures of Gaussians, under some assumptions about the public data (Bie et al., 2022; Ben-David et al., 2023) (see also (Tramèr et al., 2022)). The objective is distinct from our goal of private adaptation in supervised learning.

The most closely related work to ours is the recent study of Bassily et al. (2022), which considers a similar adaptation scenario with a public source domain and a private target domain and which also gives private algorithms with theoretical guarantees. However, that work can be distinguished from ours in several aspects. First, the authors consider a purely unsupervised adaptation scenario where no labeled sample is available from the target domain, while we consider a supervised scenario. Our study and algorithms can be extended to the unsupervised or weakly supervised setting using the notion of *unlabeled discrepancy* (Mansour et al., 2009), by leveraging upper bounds on *labeled discrepancy* in terms of unlabeled discrepancy as in (Awasthi et al., 2024). Second, the learning guarantees of our private algorithms benefit from the recent optimization of Awasthi et al. (2024), which they show have stronger learning guarantees than those of the DM solution of Cortes & Mohri (2014) adopted by Bassily et al. (2022). Similarly, in our experiments, our convex optimization solution outperforms the DM algorithm. Note that the empirical study of Bassily et al. (2022) is limited to a single artificial dataset, while we present empirical results with several non-artificial datasets. Third, our private adaptation algorithms cover regression and classification, while those of Bassily et al. (2022) only address regression with the squared loss. In Appendix A, we further discuss related work in adaptation and privacy.

We first introduce in Section 2 several basic concepts and notation for adaptation and privacy, as well as the learning problem we consider. Next, in Section 3 we describe supervised adaptation optimization by Awasthi et al. (2024) and derive private versions for that setting in Section 4 and Section 5. Finally, Section 6 provides experimental results.

2. Preliminaries

We write $\mathcal{X}$ to denote the input space and $\mathcal{Y}$ the output space which, in the regression setting, is assumed to be a measurable subset of $\mathbb{R}$. We will consider a hypothesis set $\mathcal{H}$ of functions mapping from $\mathcal{X}$ to $\mathcal{Y}$ and a loss function $\ell\colon \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$. We will denote by $B > 0$ an upper bound on the loss $\ell(h(x), y)$ for $h \in \mathcal{H}$ and $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Given a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$, we denote by $\mathcal{L}(\mathcal{D}, h)$ the expected loss of h over $\mathcal{D}$, $\mathcal{L}(\mathcal{D}, h) = \mathbb{E}_{(x,y)\sim\mathcal{D}}[\ell(h(x), y)]$.

Domain adaptation. We study a (supervised) domain adaptation problem with a public source domain defined by a distribution $\mathcal{D}^{\mathsf{Pub}}$ over $\mathcal{X} \times \mathcal{Y}$ and a private target domain defined by a distribution $\mathcal{D}^{\mathsf{Priv}}$ over $\mathcal{X} \times \mathcal{Y}$. We assume that the learner receives a labeled sample S^{Pub} of size m drawn i.i.d. from $\mathcal{D}^{\mathsf{Pub}}$, $S^{\mathsf{Pub}} = \big((x_1^{\mathsf{Pub}}, y_1^{\mathsf{Pub}}), \ldots, (x_m^{\mathsf{Pub}}, y_m^{\mathsf{Pub}})\big)$, as well as a labeled sample $\mathcal{D}^{\mathsf{Priv}}$ of size n drawn i.i.d. from $\mathcal{D}^{\mathsf{Priv}}$, $S^{\mathsf{Priv}} = \big((x_1^{\mathsf{Priv}}, y_1^{\mathsf{Priv}}), \ldots, (x_n^{\mathsf{Priv}}, y_n^{\mathsf{Priv}})\big)$. The size of the target sample n is typically more modest than that of the source sample in applications, $n \ll m$, but we will not require that assumption and will also consider alternative

scenarios. For convenience, we also write $S = \left(S^{\mathsf{Pub}}, S^{\mathsf{Priv}}\right)$ to denote the full sample of size $m+n$.

Learning scenario. The learning problem consists of using both samples to select a predictor $h \in \mathcal{H}$ with *privacy guarantees* and small expected loss with respect to the target distribution $\mathcal{D}^{\mathsf{Priv}}$. The notion of privacy we adopt is that of *differential privacy* (Dwork et al., 2006a;b; Dwork & Roth, 2014a), which in this context can be defined as follows.

Differential privacy: Given ε and $\delta > 0$, a (randomized) algorithm $\mathcal{M}\colon (\mathcal{X} \times \mathcal{Y})^{m+n} \to \mathcal{H}$ is said to be (ε, δ)-*differentially private* if for any public sample $S^{\mathsf{Pub}} \in (\mathcal{X} \times \mathcal{Y})^m$, for any pair of private datasets S^{Priv} and $\hat{S}^{\mathsf{Priv}} \in (\mathcal{X} \times \mathcal{Y})^n$ that differ in exactly one entry, and for any measurable subset $\mathcal{O} \subseteq \mathcal{H}$, we have: $\mathbb{P}\big(\mathcal{M}((S^{\mathsf{Pub}}, S^{\mathsf{Priv}})) \in \mathcal{O}\big) \leq e^{\varepsilon}\, \mathbb{P}\big(\mathcal{M}((S^{\mathsf{Pub}}, \hat{S}^{\mathsf{Priv}})) \in \mathcal{O}\big) + \delta$. Thus, the information gained by an observer is approximately invariant to the presence or absence of a sample point in the private sample.

Discrepancy. For adaptation to be successful, the source and target distributions must be close according to an appropriate divergence measure. Several notions of *discrepancy* have been shown to be adequate divergence measures in previous theoretical analyses of adaptation problems (Kifer et al., 2004; Mansour et al., 2009; Mohri & Muñoz Medina, 2012; Cortes & Mohri, 2014; Cortes et al., 2019). We will denote by $\mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}})$ the *labeled discrepancy* of $\mathcal{D}^{\mathsf{Priv}}$ and $\mathcal{D}^{\mathsf{Pub}}$, also called $\mathcal{Y}$-discrepancy in (Mohri & Muñoz Medina, 2012; Cortes et al., 2019) and defined by:

$$\mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}) = \sup_{h \in \mathcal{H}} \left| \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) \right|. \quad (1)$$

Labeled discrepancy can be straightforwardly upper bounded in terms of the $\|\cdot\|_1$ distance of the private and public distributions: $\mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}) \leq B \|\mathcal{D}^{\mathsf{Priv}} - \mathcal{D}^{\mathsf{Pub}}\|_1$. Some of its key benefits are that, unlike the $\|\cdot\|_1$-distance, it takes into account the loss function and the hypothesis set and it can be estimated from finite samples, also in the privacy preserving setting, see Section 4. Note that, while we are using absolute values for the difference of expectations, our analysis does not require that and the proofs hold with a one-sided definition. In some instances, a finer and more favorable notion of *local discrepancy* can be used, where the supremum is restricted to a subset $\mathcal{H}_1 \subset \mathcal{H}$ (Cortes et al., 2019; de Mathelin et al., 2021; Zhang et al., 2019; 2020b).

3. Optimization Problem for Supervised Adaptation

Let $\hat{d}$ denote the empirical estimate of the discrepancy based on the samples S^{Pub} and S^{Priv}:

$$\hat{d} \triangleq \sup_{h \in \mathcal{H}} \left| \frac{1}{n} \sum_{i=m+1}^{m+n} \ell(h(x_i), y_i) - \frac{1}{m} \sum_{i=1}^{m} \ell(h(x_i), y_i) \right|. \quad (2)$$

Let $\mathsf{q} \in [0,1]^{m+n}$ denote a vector of weights over the full sample $(S^{\mathsf{Pub}}, S^{\mathsf{Priv}})$, which, depending on their values, are used to emphasize or deemphasize the loss on each sample (x_i, y_i). We also denote by $\overline{\mathsf{q}}^{\mathsf{Pub}}$ the *total weight* on the first m (public) points, $\overline{\mathsf{q}}^{\mathsf{Pub}} = \sum_{i=1}^{m} \mathsf{q}_i$, and by $\overline{\mathsf{q}}^{\mathsf{Priv}}$ the *total weight* on the next n (private) ones, $\overline{\mathsf{q}}^{\mathsf{Priv}} = \sum_{i=m+1}^{m+n} \mathsf{q}_i$. Note that q is not required to be a distribution. Then, the following joint optimization problem based on a q-weighted empirical loss and the empirical discrepancy $\hat{d}$ was suggested by Awasthi et al. (2024) for supervised domain adaptation.

$$\min_{\substack{h \in \mathcal{H} \\ \mathsf{q} \in \mathcal{Q}}} \sum_{i=1}^{m+n} \mathsf{q}_i \big[\ell(h(x_i), y_i) + \hat{d} 1_{i \leq m} \big] \quad (3)$$
$$+ \lambda_1 \|\mathsf{q} - \mathsf{p}^0\|_1 + \lambda_2 \|\mathsf{q}\|_2 + \lambda_\infty \|\mathsf{q}\|_\infty,$$

where $\mathcal{Q} = [0,1]^{m+n}$ and where λ_1, λ_2 and λ_∞ are non-negative hyperparameters. Here, p^0 is a *reference* or *ideal* reweighting choice, further discussed below.

This optimization is directly based on minimizing the right-hand side of a generalization bound (see Theorem B.1, Appendix B), for which we give a self-contained and more concise proof. Moreover, (Awasthi et al., 2024) established a corresponding lower bound for any reweighting technique in terms of the q-weighted empirical loss and the discrepancy of the distributions $\mathcal{D}^{\mathsf{Priv}}$ and $\mathcal{D}^{\mathsf{Pub}}$ for a weight vector q. This further validates the significance of the generalization bound. It suggests that the optimization problem (3) admits the strongest theoretical learning guarantee we can hope for and can be regarded as an *ideal* algorithm among reweighting-based algorithms for supervised adaptation.

The key idea behind the optimization is to assign different weights q to labeled samples to account for the presence of distinct domain distributions, akin to reweighting strategies such as importance weighting. The success of adaptation hinges on a favorable balance of several crucial factors expressed by Theorem B.1. We aim to select a predictor h with a small q-weighted empirical loss ($\sum_i \mathsf{q}_i \ell(h(x_i), y_i)$). Yet, we must limit the total q-weight assigned to source domain samples if the empirical discrepancy $\hat{d}$ is substantial (captured by the $\hat{d}$ term in (3)). Avoiding disproportionate weighting on a few points is critical to maintaining an adequate effective sample size, addressed by the inclusion of the norm-2 term $\|\mathsf{q}\|_2$. The norm-1 term encourages the choice of q not deviating significantly from the reference weights p^0, while the norm-∞ term relates to controlling complexity. A careful empirical tuning of the hyperparameters helps achieve a judicious balance between these terms, leading to a well-performing adaptation process. A more detailed discussion is given in Appendix E.

A natural reference p^0, which we assume in the following, is an α-mixture of the empirical distributions $\hat{\mathcal{D}}^{\mathsf{Pub}}$ and $\hat{\mathcal{D}}^{\mathsf{Priv}}$ associated to the samples S^{Pub} and S^{Priv}: $\mathsf{p}^0 = \alpha \hat{\mathcal{D}}^{\mathsf{Pub}} + (1-\alpha) \hat{\mathcal{D}}^{\mathsf{Priv}}$, with $\alpha \in (0,1)$. Thus, p^0_i is equal to $\frac{\alpha}{m}$ if

$i \in [1, m]$, $\frac{1-\alpha}{n}$ otherwise. The mixture parameter α can be chosen as a function of the estimated discrepancy.

An important advantage of the solution based on this optimization problem is that the weights q are selected in conjunction with the predictor h. This is unlike most reweighting techniques in the literature, such as importance weighting (Sugiyama et al., 2007b; Lu et al., 2021; Sugiyama et al., 2007a; Cortes et al., 2010; Zhang et al., 2020a), KLIEP (Sugiyama et al., 2007b), Kernel Mean Matching (KMM) (Huang et al., 2006), discrepancy minimization (DM) (Cortes & Mohri, 2014), and gapBoost (Wang et al., 2019), that consist of first pre-determining some weights q irrespective of the choice of h, and subsequently selecting h by minimizing a q-weighted empirical loss.

Optimality and theoretical guarantees. In light of the theoretical properties already underscored, we define an ideal algorithm for *private* supervised adaptation as one that achieves (ϵ, δ)-DP and returns a solution closely approximating that of problem (3). This paper introduces two differentially private algorithms that precisely fulfill these criteria. We also present empirical evidence demonstrating the proximity of the performance of our private algorithms to that of the non-private optimization problem (3).

For a fixed choice of q, an additional error term of $\Omega\Big(\frac{\sqrt{d \log \frac{1}{\delta}}}{\varepsilon n}\Big)$ is necessary to ensure privacy for convex ERM (Bassily et al., 2014; Steinke & Ullman, 2015). Here, d represents the dimension of the parameter space. Therefore, the term is necessary in the expected loss of any (ϵ, δ)-differentially private supervised adaptation algorithm based on sample reweighting. Remarkably, the theoretical guarantee that we prove for our first algorithm only differs from that of (3) by a term closely matching $\frac{\sqrt{d \log \frac{1}{\delta}}}{\varepsilon n}$.

4. Private Adaptation Algorithm for Regression with Linear Predictors

In this section, we consider a regression problem with the squared loss and using linear predictors, for which we give a differentially private adaptation algorithm.

We consider an input space $\mathcal{X} = \big\{x \in \mathbb{R}^d \colon \|x\|_2 \leq r\big\}$, $r > 0$, an output space $\mathcal{Y} = \{y \in \mathbb{R} \colon |y| \leq 1\}$, and a family of bounded linear predictors $\mathcal{H} = \{x \mapsto w \cdot x \mid w \in \mathcal{W}\}$, where $\mathcal{W} = \{w \colon \|w\|_2 \leq \Lambda\}$, for some $\Lambda > 0$. This covers scenarios where we fix lower layers of a pre-trained neural network and only seek to learn the parameters of the top layer.

Note that the squared loss is bounded for $x \in \mathcal{X}$ and $w \in \mathcal{W}$: $\ell_{\mathsf{sq}}(w \cdot x, y) = (w \cdot x - y)^2 \leq (\Lambda r + 1)^2 \triangleq B$. It is also G-Lipschitz, $G \triangleq 2r(\Lambda r + 1)$, with respect to $w \in \mathcal{W}$ since $|\ell_{\mathsf{sq}}(w \cdot x, y) - \ell_{\mathsf{sq}}(w' \cdot x, y)| \leq G\|w - w'\|_2$ for all $w, w' \in \mathcal{W}$ and $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Furthermore, $\ell_{\mathsf{sq}}(\cdot, y)$ is convex in $y \in \mathcal{Y}$.

To devise an (ϵ, δ)-differentially private algorithm for our adaptation scenario, we first show how to privately estimate the discrepancy term. Next, we show that the natural optimization problem (3) can be cast as a convex optimization problem, for which we design a favorable private solution.

Discrepancy estimates. Since ℓ_{sq} is convex with respect to its first argument, problem (2) can be cast as two difference-of-convex problems ((DC)-programming) by removing the absolute value and considering both possible signs. Each of these problems can be solved using the DCA algorithm of (Tao & An, 1998) (see also (Yuille & Rangarajan, 2003; Sriperumbudur et al., 2007)). Furthermore, for the squared loss ℓ_{sq} with our linear hypotheses, the DCA method is guaranteed to reach a global optimum (Tao & An, 1998).

Now, observe that the empirical discrepancy $\hat{d}$ defined in (2) is estimated using private data, therefore, the addition of noise is crucial to ensure the differential privacy (DP). This term solely depends on the dataset S and not on the particular choice of h or q and it remains unchanged through the iterations of Algorithm 1. Therefore, its (private) estimation can be performed upfront, prior to the algorithm's execution. Furthermore, the sensitivity of $\hat{d}$ with respect to replacing one point in S^{Priv} is at most B/n. Thus, we can first generate an $\epsilon/2$-differentially private version $\hat{d}_{\mathrm{DP}} = \mathsf{Proj}_{[0,B]}\big(\hat{d} + \nu\big)$ of $\hat{d}$ by augmenting $\hat{d}$ with $\nu \sim \mathsf{Lap}(2B/(\epsilon n))$, where $\mathsf{Lap}(2B/(\epsilon n))$ is a Laplace distribution with scale $2B/(\epsilon n)$, and projecting over the interval $[0, B]$. Thus, we modify the objective function (3) by replacing $\hat{d}$ with $\hat{d}_{\mathrm{DP}}$. Note also that by the properties of the Laplace distribution and the fact that $\hat{d} \geq 0$, the expected excess optimization error due to this modification is in $O(B/(\epsilon n))$, which, as we shall see, is dominated by the excess error due to privately optimizing the new objective. In the following, in the private optimization algorithm we present (Algorithm 1), we instantiate the privacy parameter with $\varepsilon/2$ so that the overall algorithm is (ε, δ)-differentially private. Alternatively, we could add noise directly to the gradients to account for the empirical discrepancy term estimated from the private data. However, a straightforward analysis shows that this approach would introduce significantly more noise to the gradients.

After substituting with our choice of a uniform reference distribution, as mentioned in Section 2, the problem reduces to privately solving the following optimization problem:

$$\min_{\substack{\|w\|_2 \leq \Lambda \\ \mathsf{q} \in \mathcal{Q}}} \sum_{i=1}^{m+n} \mathsf{q}_i \Big[(w \cdot x_i - y_i)^2 + \hat{d}_{\mathrm{DP}} 1_{i \leq m} \Big] \tag{4}$$
$$+ \lambda_1 \left[\sum_{i=1}^{m} \left| \mathsf{q}_i - \frac{\alpha}{m} \right| + \sum_{i=m+1}^{n+m} \left| \mathsf{q}_i - \frac{1-\alpha}{n} \right| \right] + \lambda_2 \|\mathsf{q}\|_2 + \lambda_\infty \|\mathsf{q}\|_\infty .$$

This optimization presents two main challenges: (1) while it is convex with respect to w and with respect to q, it is

not jointly convex; (2) the gradient sensitivity with respect to q of the objective is a constant and thus not favorable to derive differential privacy guarantees. In the following, we will show how both issues can be tackled. Inspecting (4) leads to the following useful observation.

Observe that for each $i \in [m]$, the objective is increasing in q_i over $\left[\frac{\alpha}{m}, 1\right]$ and similarly, for each $i \in [m+1, m+n]$, it is increasing in q_i over the interval $\left[\frac{1-\alpha}{n}, 1\right]$. Thus, the following stricter constraints on q in (4) do not affect the optimal solution: $\forall i \in [m],\ 0 \le \mathsf{q}_i \le \frac{\alpha}{m}; \forall i \in [n],\ 0 \le \mathsf{q}_{m+i} \le \frac{1-\alpha}{n}$. The problem can thus be equivalently formulated as:

$$\min_{\|w\|_2 \le \Lambda, \mathsf{q}} \sum_{i=1}^{m+n} \mathsf{q}_i \Big[(w \cdot x_i - y_i)^2 + \hat{d}_{\mathrm{DP}} 1_{i \le m} \Big] + \lambda_1 \Big[1 - \sum_{i=1}^{m+n} \mathsf{q}_i \Big] + \lambda_2 \|\mathsf{q}\|_2 + \lambda_\infty \|\mathsf{q}\|_\infty \tag{5}$$
$$\text{s.t.} \quad \forall i \in [m],\ 0 \le \mathsf{q}_i \le \frac{\alpha}{m}; \forall i \in [n],\ 0 \le \mathsf{q}_{n+i} \le \frac{1-\alpha}{n}.$$

Convex-optimization formulation. We now derive a convex formulation of this optimization problem, which enables us to devise an efficient private algorithm with formal convergence guarantee. We introduce new variables $\mathsf{u}_i = \frac{1}{\mathsf{q}_i}$, $\forall i \in [1, m+n]$, and use the following upper bound that holds by the convexity of $x \mapsto 1/x$ on $\mathbb{R}_+^*$:

$$1 - \sum_{i=1}^{m+n} \frac{1}{\mathsf{u}_i} \le \left(\frac{\alpha}{m}\right)^2 \sum_{i=1}^{m} \mathsf{u}_i + \left(\frac{1-\alpha}{n}\right)^2 \sum_{i=m+1}^{m+n} \mathsf{u}_i - 1.$$

This yields the following optimization problem in (w, u):

$$\min_{\|w\|_2 \le \Lambda, \mathsf{u}} \sum_{i=1}^{m+n} \frac{(w \cdot x_i - y_i)^2 + \hat{d}_{\mathrm{DP}}\, 1_{i \le m}}{\mathsf{u}_i} \tag{6}$$
$$+ \kappa_1 \left[\frac{\alpha^2}{m^2} \sum_{i=1}^{m} \mathsf{u}_i + \frac{(1-\alpha)^2}{n^2} \sum_{i=m+1}^{m+n} \mathsf{u}_i - 1 \right] + \kappa_2 \left[\sum_{i=1}^{m+n} \frac{1}{\mathsf{u}_i^2} \right]^{\frac{1}{2}} + \frac{\kappa_\infty}{\min_i \mathsf{u}_i}$$
$$\text{s.t.} \quad \forall i \in [m], \mathsf{u}_i \ge \frac{m}{\alpha};\ \forall i \in [n], \mathsf{u}_{m+i} \ge \frac{n}{1-\alpha},$$

with new hyperparameters $\kappa_1, \kappa_2, \kappa_\infty$. The problem (6) is a joint convex optimization problem in w and u since the constraints on u are affine, the constraint on w is convex, and since each term $\frac{(w \cdot x_i - y_i)^2}{\mathsf{u}_i}$ is jointly convex in (w, u_i) as a quadratic-over-linear function (Boyd & Vandenberghe, 2014). We will denote by $\mathsf{F}(w, \mathsf{u})$ the objective function and by $\mathcal{U}$ the feasible set of u, $\mathcal{U} = \left([\frac{m}{\alpha}, \infty)^m \times [\frac{n}{1-\alpha}, \infty)^n \right)$.

We can establish that, under a reasonable assumption, the incremental error resulting from optimizing problem (6) based on the convex upper bound we used, instead of optimizing the original objective (5) remains negligible. Specifically, we can show that this additional error is bounded by $O(\|\mathsf{q}^* - \mathsf{p}^0\|_1)$, where q^* represents the optimal weight vector for the original (5) and p^0 denotes the reference distribution. We expect this term, $O(\|\mathsf{q}^* - \mathsf{p}^0\|_1)$, to be much smaller than $O(1/\sqrt{n})$ in all scenarios where the bound of Theorem B.1 (Appendix B) is advantageous, that is, when it provides a benefit over training solely on the target sample. We formally state this in the theorem below, with the proof provided in Appendix C.1.

Theorem 4.1. *Let $\mathsf{G}(w, \mathsf{q})$ denote the objective function of problem (5) and (w^*, q^*) its minimizer (the solution of (5)). Let $(\tilde{w}, \tilde{\mathsf{u}})$ be the minimizer of $\mathsf{F}(w, \mathsf{u})$ (the solution of (6)). Define $\tilde{\mathsf{q}} \in [0,1]^{m+n}$ as the weight vector obtained by applying the inverse transformation to $\tilde{\mathsf{u}}$: $\tilde{\mathsf{q}}_i \to \frac{1}{\tilde{\mathsf{u}}_i}, \forall i \in [m+n]$. Assume that there exists a universal constant $C \ge 1$ such that $\forall i \in [m+n], C\mathsf{q}_i^* \ge \mathsf{p}_i^0$. Then, then the following inequality holds:*

$$\mathsf{G}(\tilde{w}, \tilde{\mathsf{q}}) \le \mathsf{G}(w^*, \mathsf{q}^*) + O(\|\mathsf{p}^0 - \mathsf{q}^*\|_1).$$

In the following, we will use the shorthands $\mathsf{u}^{\mathsf{Pub}} = (\mathsf{u}_1, \ldots, \mathsf{u}_m)$ and $\mathsf{u}^{\mathsf{Priv}} = (\mathsf{u}_{m+1}, \ldots, \mathsf{u}_{m+n})$, and denote by $\widehat{L}^{\mathsf{Pub}}(w, \mathsf{u}^{\mathsf{Pub}})$ and $\widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$ their contributions to the empirical loss: $\widehat{L}^{\mathsf{Pub}}(w, \mathsf{u}^{\mathsf{Pub}}) \triangleq \sum_{i=1}^{m} \frac{(w \cdot x_i^{\mathsf{Pub}} - y_i^{\mathsf{Pub}})^2 + \hat{d}_{\mathrm{DP}}}{\mathsf{u}_i^{\mathsf{Pub}}}$, $\widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}}) \triangleq \sum_{i=1}^{n} \frac{(w \cdot x_i^{\mathsf{Priv}} - y_i^{\mathsf{Priv}})^2}{\mathsf{u}_i^{\mathsf{Priv}}}$. Note that, since differential privacy is closed under postprocessing, $\hat{d}_{\mathrm{DP}}$ is safe to publish as a differentially private estimate of $\hat{d}$. Thus, the only term in $\mathsf{F}(w, \mathsf{u})$ that is sensitive from a privacy perspective is $\widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$.

Let $\nabla_w, \nabla_{\mathsf{u}^{\mathsf{Pub}}}$, and $\nabla_{\mathsf{u}^{\mathsf{Priv}}}$ denote the gradients with respect to $w, \mathsf{u}^{\mathsf{Pub}}$, and $\mathsf{u}^{\mathsf{Priv}}$, respectively. We will also use the shorthand $\bar{B} \triangleq B + \kappa_1 + \kappa_2 + \kappa_\infty$. The following lemma shows several important properties of the objective function. The proof is given in Appendix C.2.

Lemma 4.2. *The following properties hold for the objective function F.*

(i) The following upper bounds hold for the gradients, for all $(w, \mathsf{u}) \in \mathcal{W} \times \mathcal{U}$: $\|\nabla_w \mathsf{F}(w, \mathsf{u})\|_2 \le G$, $\|\nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w, \mathsf{u})\|_2 \le \frac{\alpha^2 (B + \bar{B})}{m^{3/2}}$, and $\|\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w, \mathsf{u})\|_2 \le \frac{(1-\alpha)^2 \bar{B}}{n^{3/2}}$.

(ii) The ℓ_2-sensitivity of $\nabla_w \widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$ with respect to changing one private data point is at most $\frac{2(1-\alpha)G}{n}$;
(iii) The ℓ_2-sensitivity of $\nabla_{\mathsf{u}^{\mathsf{Priv}}} \widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$ with respect to changing one private data point is at most $\frac{(1-\alpha)^2 B}{n^2}$.

Algorithm 1, denoted CnvxAdap, gives pseudocode for our differentially private adaptation algorithm based on the convex problem (6). Our algorithm is a variant of noisy projected gradient descent with steps 7, 9, and 11 implementing the Euclidean projection of (w_t, u_t) onto the constraint set $\mathcal{W} \times \mathcal{U}$. The non-private version of Algorithm 1 we will denote $\mathsf{CnvxAdap}_\infty$ for $\epsilon = \infty$. The following provides both a DP and convergence guarantee for our algorithm.

Theorem 4.3. *For any $\delta > 0$ and $0 < \varepsilon \le 8\log(1/\delta)$, Algorithm 1 is (ε, δ)-differentially private. Furthermore, let*

Algorithm 1 CnvxAdap Private adaptation algorithm based on F

Require: $S^{\mathsf{Pub}} \in (\mathcal{X} \times \mathcal{Y})^m$; $S^{\mathsf{Priv}} \in (\mathcal{X} \times \mathcal{Y})^n$; privacy parameters (ε, δ); hyperparameters $\kappa_1, \kappa_2, \kappa_\infty$; number of iterations T.

Choose (w_0, u_0) in $\mathcal{W} \times \mathcal{U}$ arbitrarily.
Set $\sigma_1 := \frac{2s_1\sqrt{T\log(\frac{3}{\delta})}}{\varepsilon}$, where $s_1 := \frac{2(1-\alpha)G}{n}$.
Set $\sigma_2 := \frac{2s_2\sqrt{T\log(\frac{3}{\delta})}}{\varepsilon}$, where $s_2 := \frac{(1-\alpha)^2 B}{n^2}$.
Set step sizes $\eta_w := \frac{\Lambda}{\sqrt{T\left(G^2+d\sigma_1^2\right)}}$, $\eta_{\mathsf{u}^{\mathsf{Pub}}} := \frac{m^{3/2}}{\sqrt{T}\,\alpha^2(B+\bar{B})}$, and $\eta_{\mathsf{u}^{\mathsf{Priv}}} := \frac{n^{3/2}}{\sqrt{T\left((1-\alpha)^4\bar{B}^2+n^4\sigma_2^2\right)}}$.
for $t = 0$ to $T-1$ **do**
  $w_{t+1} := w_t - \eta_w(\nabla_w \mathsf{F}(w_t, \mathsf{u}_t) + \mathbf{z}_t)$, where $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \sigma_1^2 \mathbb{I}_d)$.
  If $\|w_{t+1}\|_2 > \Lambda$ then $w_{t+1} \leftarrow \Lambda \frac{w_{t+1}}{\|w_{t+1}\|_2}$.
  $\mathsf{u}_{t+1}^{\mathsf{Pub}} := \mathsf{u}_t^{\mathsf{Pub}} - \eta_{\mathsf{u}^{\mathsf{Pub}}} \nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w_t, \mathsf{u}_t)$.
  For every $i \in [m]$, set $\mathsf{u}_{i,t+1}^{\mathsf{Pub}} \leftarrow \max\left(\mathsf{u}_{i,t+1}^{\mathsf{Pub}}, \frac{m}{\alpha}\right)$.
  $\mathsf{u}_{t+1}^{\mathsf{Priv}} := \mathsf{u}_t^{\mathsf{Priv}} - \eta_{\mathsf{u}^{\mathsf{Priv}}}\left(\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w_t, \mathsf{u}_t) + \mathbf{z}_t'\right)$, where $\mathbf{z}_t' \sim \mathcal{N}(\mathbf{0}, \sigma_2^2 \mathbb{I}_n)$.
  For every $i \in [n]$, set $\mathsf{u}_{i,t+1}^{\mathsf{Priv}} \leftarrow \max\left(\mathsf{u}_{i,t+1}^{\mathsf{Priv}}, \frac{n}{1-\alpha}\right)$.
end for
return $(\bar{w}, \bar{\mathsf{u}}) = \frac{1}{T}\sum_{t=1}^{T}(w_t, \mathsf{u}_t)$.

(w^*, u^*) be a minimizer of $\mathsf{F}(w, \mathsf{u})$, then, the expected optimization error of the solution $(\bar{w}, \bar{\mathsf{u}})$ returned by Algorithm 1 is bounded as follows:

$$\mathsf{F}(\bar{w}, \bar{\mathsf{u}}) - \mathsf{F}(w^*, \mathsf{u}^*) \leq O\left(\frac{G\Lambda\sqrt{d\log\frac{1}{\delta}}}{n\varepsilon} + \frac{B\max\left(1, \|\mathsf{u}_0 - \mathsf{u}^*\|_2^2\right)\sqrt{\log\frac{1}{\delta}}}{n^{3/2}\varepsilon}\right),$$

for $T \geq \max\left(1, \frac{n^2\varepsilon^2}{d(1-\alpha)^2\log(\frac{1}{\delta})}, \frac{\bar{B}^2\varepsilon^2}{B^2\log(\frac{1}{\delta})}, \frac{\varepsilon^2\bar{B}^2 n^3}{\log(\frac{1}{\delta})B^2 m^3}\right)$.

The more explicit form of the bound as well as the proof are presented in Appendix C.3. We note here that, for sufficiently large n, our bound scales as $O\left(G\Lambda\sqrt{d\log(1/\delta)}/n\varepsilon\right)$. This bound on the optimization error is essentially optimal for our optimization problem under differential privacy. To see this, note that our optimization task is generally harder than the standard empirical risk minimization (ERM) as the latter task can be viewed as a simple instantiation of our optimization problem, where the optimal weights q are uniform on the private data and zero on the public data, and they are given to the algorithm beforehand (hence, the optimization algorithm is only required to optimize over the parameters vector w). Hence, the known lower bound of $\Omega\left(G\Lambda\sqrt{d\log(1/\delta)}/n\varepsilon\right)$ on private convex ERM (Bassily et al., 2014)[1] implies a lower bound on the optimization error in our problem. Note that this implies that the additional error incurred by our algorithm due to privacy (i.e., compared to the best non-private algorithm for optimizing (3)) scales as $O\left(\sqrt{d\log(1/\delta)}/\varepsilon n\right)$, which matches the necessary additional error (due to privacy) in the expected loss of any private algorithm based on sample reweighting, as discussed in Section 3. Formally stated:

Theorem 4.4. *Suppose $n \geq \frac{\left(B\max\left(1, U^{\mathsf{Pub}}+U^{\mathsf{Priv}}\right)\right)^2}{(G\Lambda)^2 d}$ and $T \geq \max\left(1, \frac{n^2\varepsilon^2}{d(1-\alpha)^2\log(\frac{1}{\delta})}, \frac{\bar{B}^2\varepsilon^2}{B^2\log(\frac{1}{\delta})}, \frac{\varepsilon^2\bar{B}^2 n^3}{\log(\frac{1}{\delta})B^2 m^3}\right)$ in Algorithm 1. Then, the resulting expected optimization error is $O\left(\frac{G\Lambda\sqrt{d\log(1/\delta)}}{n\varepsilon}\right)$, which is optimal.*

5. Private Adaptation – More General Settings

Here, we consider a general possibly non-convex setting, where the loss function is only assumed to be G-Lipschitz and β-smooth, that is differentiable, with β-Lipschitz gradient in the parameter w with respect to the ℓ_2-norm. To simplify, we will here abusively adopt the notation $\ell(w, x, y)$ to denote the loss associated with a parameter vector $w \in \mathcal{W}$ (defining a hypothesis) and a labeled point $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Since the ℓ_2-diameter of $\mathcal{W}$ is Λ-bounded and the loss is Lipschitz, we assume without loss of generality that ℓ is uniformly bounded by $B = G\Lambda$ over $\mathcal{W} \times \mathcal{X} \times \mathcal{Y}$.

Our goal is to privately optimize a more general version of problem 5, where the squared loss is replaced with any such loss ℓ. This is in general a non-convex optimization problem and finding a global minimum is hence intractable. An alternative, widely adopted in the literature, is to find a stationary point of the non-convex objective.

Challenges for the design of a private adaptation algorithm. Before we discuss the stationarity criterion, we first describe our approach. One issue with the objective when expressed as a function of $\mathsf{q} = (\mathsf{q}^{\mathsf{Pub}}, \mathsf{q}^{\mathsf{Priv}})$ is that its gradient with respect to $\mathsf{q}^{\mathsf{Priv}}$ admits an $\Omega(1)$ sensitivity. That can improved by introducing new variables $\tilde{\mathsf{q}}^{\mathsf{Pub}} = \frac{\alpha}{m}\mathsf{q}^{\mathsf{Pub}}$ and $\tilde{\mathsf{q}}^{\mathsf{Priv}} = \frac{1-\alpha}{n}\mathsf{q}^{\mathsf{Priv}}$, thereby reducing the sensitivity of the gradient with respect to $\tilde{\mathsf{q}}^{\mathsf{Priv}}$ to $O\left(\frac{1}{n}\right)$. Since this gradient is n-dimensional, the magnitude of the noise added for privacy will be $O\left(\frac{1}{\sqrt{n}}\right)$. However, we can further enhance the sensitivity if we resort to the reparameterization technique described in Section 4. As we show in the sequel, by applying the transformation of variables $\mathsf{u}_i = \frac{1}{\mathsf{q}_i}$, $i \in [m+n]$, we are able to reduce the sensitivity of the gradient components, and hence achieve better convergence guarantees.

[1] The lower bound in (Bassily et al., 2014) does not have the $\sqrt{\log(1/\delta)}$ factor, but it's been known that the lower bound can be improved to include this factor via the more recent results of (Steinke & Ullman, 2015).

Another challenge we face here is the non-smoothness of the objective. Together with the non-convex functions, it leads to weaker convergence guarantees even in the non-private setting (Arjevani et al., 2019; Shamir, 2020). Although the aforementioned transformation ensures that the term corresponding to $\|\mathsf{q}\|_2$ is smooth as a function of u, the term corresponding to $\|\mathsf{q}\|_\infty$, i.e., $\frac{1}{\min_{i\in[m+n]}\mathsf{u}_i}$, remains non-smooth. To address this issue, we replace that term with its μ-softmax approximation, namely, $\frac{1}{\mu}\log\big(\sum_{i=1}^{m+n} e^{\mu/\mathsf{u}_i}\big)$, where $\mu > 0$ is the softmax approximation parameter. A basic fact about this μ-softmax approximation is that its approximation error is uniformly bounded by $O\big(\frac{1}{\mu}\log(m+n)\big)$. With $\mu = O\big(\sqrt{m+n}\big)$, the excess error we incur is $\widetilde{O}\big(1/\sqrt{m+n}\big)$.

Thus, our goal is to privately find a stationary point of the following optimization problem:

$$\min_{\substack{\|w\|_2\le\Lambda\\ \mathsf{u}}} \sum_{i=1}^{m+n} \frac{\ell(w,x_i,y_i) + \hat{d}_{\mathrm{DP}}\,1_{i\le m}}{\mathsf{u}_i} \tag{7}$$
$$+\lambda_1\left[1-\sum_{i=1}^{m+n}\frac{1}{\mathsf{u}_i}\right] + \lambda_2\left[\sum_{i=1}^{m+n}\frac{1}{\mathsf{u}_i^2}\right]^{\frac{1}{2}} + \frac{\lambda_\infty}{\mu}\log\left[\sum_{i=1}^{m+n} e^{\frac{\mu}{\mathsf{u}_i}}\right]$$
$$\text{s.t.}\quad \forall i\in[m], \mathsf{u}_i \ge \frac{m}{\alpha};\ \forall i\in[n], \mathsf{u}_{m+i} \ge \frac{n}{1-\alpha}.$$

We denote the objective in (7) by $\mathsf{J}(w,\mathsf{u})$, and let $\mathcal{V} \triangleq \mathcal{W}\times\mathcal{U}$, where $\mathcal{W}\times\mathcal{U}$ is the constraint set in the above problem.

In the following lemma, we show several useful properties of J that will be crucial for proving the privacy and convergence guarantees of our private algorithm. We stress that our reparameterization idea, which led to the above problem formulation, was key to ensuring such properties. That and the following lemma are crucial results enabling us to devise a private adaptation solution with guarantees for this general non-convex setting. The proof is given in Appendix D.1.

Lemma 5.1. *The objective function* J *admits the following properties:*
(i) J *satisfies the same properties as those stated for* F *in Lemma 4.2.*
(ii) Assume $m^{\frac{1}{3}} = O(n), n = O(m^3)$, *and* $\mu = O\big((m+n)^{\frac{2}{3}}\big)$. *Then,* J *is* $\bar{\beta}$*-smooth over* $\mathcal{V}$, *where* $\bar{\beta} = O(\beta)$.

Gradient mapping as a stationarity criterion. Here, we adopt a standard stationarity criterion given by the norm of the *gradient mapping* (Beck, 2017). The γ-gradient mapping of a function $f\colon\mathcal{V}\to\mathbb{R}$ at $\mathsf{v}\in\mathcal{V}$ is denoted and defined by $\mathcal{G}_{f,\gamma}(\mathsf{v}) \triangleq \gamma\big(\mathsf{v} - \mathrm{Proj}_{\mathcal{V}}\big(\mathsf{v} - \frac{1}{\gamma}\nabla f(\mathsf{v})\big)\big)$. A point $\mathsf{v}^*\in\mathcal{V}$ is a stationary point of f if and only if $\|\mathcal{G}_{f,\gamma}(\mathsf{v}^*)\|_2 = 0$. For an r-smooth function f, $\|\mathcal{G}_{f,r}(\cdot)\|_2$ serves as a measure of convergence to a stationary point (Beck, 2017; Ghadimi et al., 2016; Li & Li, 2018).

Private algorithm for general adaptation settings: Our private algorithm for the non-convex problem, denoted by NCnvxAdap, is a variant of noisy projected gradient descent. Our algorithm admits exactly the same generic description as Algorithm 1, including the settings of the noise variances σ_1^2 and σ_2^2, with the following differences: (1) we use J instead of F; (2) we set $\eta_w = \eta_{\mathsf{u}^{\mathsf{Pub}}} = \eta_{\mathsf{u}^{\mathsf{Priv}}} = \eta \triangleq \frac{1}{\bar{\beta}}$, where $\bar{\beta}$ is the smoothness parameter given in Lemma 5.1; (3) the algorithm returns $(w_{t^*}, \mathsf{u}_{t^*})$, where t^* is drawn uniformly from $[T]$. The full pseudocode of NCnvxAdap is given in Appendix D.2. We will denote by $\mathsf{NCnvxAdap}_\infty$ the non-private version of NCnvxAdap.

Our algorithm is similar to that of Wang & Xu (2019), but our analysis is distinct and yields stronger guarantee than their Theorem 2. The convergence measure in that reference is based on the norm of the noisy projected gradient. In contrast, our convergence guarantee is based on the noiseless gradient mapping $\mathcal{G}_{\mathsf{J},\bar{\beta}}(w_t,\mathsf{u}_t)$, which we view as more relevant. The following theorem gives privacy and convergence guarantees for our algorithm, see proof in Appendix D.3; it establishes the total number of iterations (gradient updates) of our algorithm is in $O\big(\varepsilon/\sqrt{d\log(1/\delta)}\big)$.

Theorem 5.2. *Algorithm* NCnvxAdap *is* (ε,δ)*-differentially private. Moreover, for the choice* $T = O\big(\varepsilon n/\sqrt{d\log(1/\delta)}\big)$, *the output of the algorithm satisfies the following bound on the norm of the gradient mapping:* $\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(w_{t^*},\mathsf{u}_{t^*})\|_2^2 = O\big(\sqrt{\bar{\beta}d\log(1/\delta)}/\varepsilon n\big)$.

6. Experiments

We report here the results of several experiments for both our convex private adaptation algorithm for regression, CnvxAdap introduced in Section 4, and our non-convex private algorithm for more general settings, NCnvxAdap introduced in Section 5. Since state-of-the-art supervised domain adaptation algorithms have not been extended to the private supervised domain adaptation we consider, we adopt the following experimental validation procedure. First, we compare the non-private versions of our algorithms with the state-of-the-art supervised domain adaptation algorithm SBest by Awasthi, Cortes, and Mohri (2024) and demonstrate similar performance. Next, we show that the private versions of our algorithms perform close to their non-private counterparts for larger sample sizes or larger ε.

Non-private comparison to baselines in regression. We consider five regression datasets with dimensions as high as 384 from the UCI machine learning repository (Dua & Graff, 2017), the `Wind`, `Airline`, `Gas`, `News` and `Slice`. Detailed information about the sample sizes of these datasets is given in Table 4 (Appendix E). We compare our non-private convex algorithm $\mathsf{CnvxAdap}_\infty$ with the SBest algorithm (Awasthi et al., 2024), which has demonstrated superior performance to other supervised domain adaptation algorithms. We carry out model selection on the target validation set

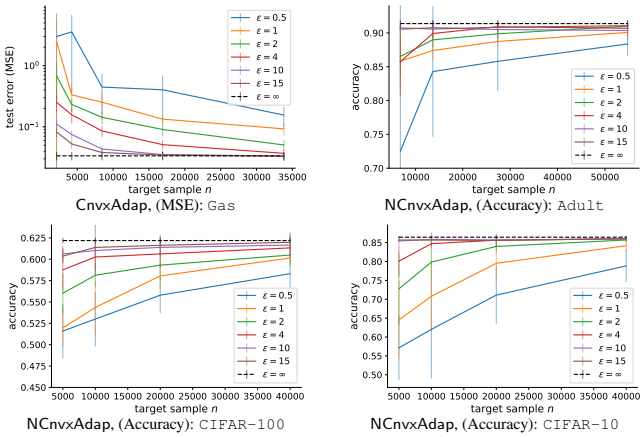

Figure 1. Performance of CnvxAdap and NCnvxAdap on various dataset against number of target samples for various values of ϵ.

and report in Table 1 the mean and standard deviation on the test set over 10 random splits of the target training and validation sets. For details on hyperparameter tuning see Appendix E. We report relative MeanSquaredErrors, MSE, normalized so that training on the target data only has an MSE of 1.0. Thus, MSE numbers less than one signify performance improvements achieved by leveraging the source data in the algorithm. The results show that our convex algorithm achieves comparable performance with the SBest algorithm. For reference, we also compare with unsupervised domain adaptation baselines such as Kernel Mean Matching (KMM) (Huang et al., 2006) and the DM algorithm (Cortes & Mohri, 2014), see Table 3 (Appendix E).

Table 1. MSE of non-private convex algorithm against baselines. We report relative errors normalized so that training on target only has an MSE of 1.0. The best results are indicated in boldface.

Dataset	SBest	CnvxAdap$_\infty$
Wind	$\mathbf{0.985 \pm 0.018}$	$\mathbf{0.985 \pm 0.019}$
Airline	$\mathbf{0.979 \pm 0.016}$	0.992 ± 0.012
Gas	0.374 ± 0.023	$\mathbf{0.342 \pm 0.023}$
News	0.996 ± 0.017	$\mathbf{0.995 \pm 0.019}$
Slice	0.995 ± 0.063	$\mathbf{0.992 \pm 0.050}$

Comparison of private and non-private algorithm in regression. Having demonstrated the quality of our non-private algorithm CnvxAdap$_\infty$, we study the performance and convergence properties of CnvxAdap as a function of the privacy guarantee ϵ and training sample size n. To obtain larger training set sizes, we sample the data with replacement (see Appendix E for a more detailed discussion). For all experiments, the privacy parameter δ is set to be 0.01.

Figure 1(upper left panel) presents the MSE on Gas over ten runs against the number of target samples with various values of ε as small as 0.5. The dashed line corresponds to the non-private convex solution of CnvxAdap$_\infty$, and is essentially flat. As we obtain larger training set sizes by sampling the data with replacement, for a point in the target data, the decrease in its weight and the increase in its frequency would largely cancel each other out in the non-private setting. We observe that the performance of our private adaptation algorithm CnvxAdap approaches that of the non-private solution ($\varepsilon = \infty$), as ε increases and as n increases, thereby verifying our convergence and theoretical analyses. For $\varepsilon = 10$ or $\varepsilon = 15$, the performance of CnvxAdap is close to that of CnvxAdap$_\infty$ for $n = 10{,}000$.

Table 2. Accuracy of non-private non-convex algorithm against supervised baselines. Best results are boldfaced.

Dataset	Train source	Train target	SBest	NCnvxAdap$_\infty$
Adult	88.44 ± 0.61	91.09 ± 0.54	91.17 ± 0.66	$\mathbf{91.38 \pm 0.49}$
German	74.16 ± 1.64	76.40 ± 1.17	77.78 ± 0.95	$\mathbf{77.96 \pm 1.31}$
Accent	51.60 ± 1.09	92.27 ± 1.10	93.61 ± 2.00	$\mathbf{93.95 \pm 0.78}$
CIFAR-100	58.50 ± 0.30	59.51 ± 0.38	$\mathbf{62.33 \pm 0.56}$	62.19 ± 0.32
CIFAR-10	83.18 ± 0.47	85.18 ± 0.24	$\mathbf{86.93 \pm 0.30}$	86.44 ± 0.29
SVHN	84.16 ± 1.21	86.28 ± 0.44	$\mathbf{87.47 \pm 0.45}$	87.34 ± 0.46
ImageNet	71.20 ± 1.75	85.70 ± 0.29	$\mathbf{87.09 \pm 0.28}$	86.84 ± 0.38

Appendix E contains additional experiments and comparisons. We compared CnvxAdap with the noisy minibatch SGD from (Bassily et al., 2019), Figure 2. We verify that our algorithm outperforms minibatch SGD, which only benefits from target labeled data. In Table 5, we report results comparing our private adaptation algorithm CnvxAdap to the non-private DM algorithm (Cortes & Mohri, 2014) on the multi-domain sentiment analysis dataset (Blitzer et al., 2007). For $\epsilon = 4$ the performance of CnvxAdap is on par with the DM algorithm, and for as low a value as $\epsilon = 10$, it clearly outperforms DM. Finally, despite the difference in scenario, as discussed in Section 1, in Appendix E we also compare our algorithm to the private-DM (Bassily et al., 2022) and show that for low values of ε and even high values of n, the private-DM does not outperform our algorithm.

Non-private comparison to baselines in classification. For the general non-convex setting, we experiment with logistic regression classifiers and consider seven datasets. Three datasets are from the UCI machine learning repository (Dua & Graff, 2017), the Adult, German and Accent (see Appendix E for the details). We also convert the CIFAR-100, CIFAR-10 (Krizhevsky, 2009), SVHN (Netzer et al., 2011) and ImageNet (Deng et al., 2009) datasets into domain adaptation tasks by establishing two distinct sampling methods for selecting the source and target data (see Appendix E for more details). In Table 2, we first compare our non-private non-convex algorithm NCnvxAdap$_\infty$ with the SBest algorithm (Awasthi et al., 2024) and report mean and standard deviation of the accuracy on the test set over 10 random splits of the target training (70%), validation (20%) and test sets (10%). The results show that our non-convex algorithm achieves comparable performance with SBest. See also Appendix E, Table 6 for a comparison with the KMM algorithm (Huang et al., 2006)).

Comparison of private and non-private algorithm in classification. We then study the performance and conver-

gence properties of NCnvxAdap as a function of the privacy guarantee ϵ and training sample size n. Figure 1 presents 3 plots of the accuracy on the `Adult`, `CIFAR-100`, and `CIFAR-10` datasets over ten runs against the number of target samples with various values of ε. The dashed line corresponds to the non-private solution of $\mathsf{NCnvxAdap}_\infty$. We observe that the performance of our private algorithm approaches that of the non-private solution ($\varepsilon = \infty$), as ε increases and as n increases. For $\varepsilon = 10$ or $\varepsilon = 15$, the performance is close to the dashed line for $n = 10{,}000$.

7. Discussion

Our proof and algorithmic techniques, including our novel reparameterization method, offer potential tools for the analysis of other related problems. We briefly highlight three potential extensions in the following discussion.

Reverse scenario. We presented a detailed analysis of differentially private learning algorithms with strong theoretical guarantees for adaptation from a public source domain to a private domain. An equally important scenario is that of differentially private adaptation from a private source domain to a public target domain. This problem arises in various crucial applications. For example, in healthcare data sharing, a hospital may wish to publish aggregated statistics about patient outcomes to contribute to public health research without compromising patient privacy. In Census data release, a national statistical agency might want to release census data for public use in research, policy-making, and business planning. For smart grid data, utility companies collect detailed electricity usage data from smart meters, which is private, to learn about enhanced grid management and improve energy efficiency. Similarly, in location-based services, a company providing services such as a navigation app may wish to release aggregate data on traffic patterns and points of interest to improve urban planning and business strategies.

On online platforms and social media, a social media platform may aim to share insights on user engagement and content popularity trends with marketers and researchers. For public safety and crime statistics, law enforcement agencies collect crime data and aim to share this information to inform public safety initiatives and academic research. Lastly, in educational research, institutions collect data on student performance and learning behaviors, which may be used to improve educational strategies and outcomes.

Given the structure of the learning bound and the objective function in our private optimization-based adaptation algorithms, our algorithms can be straightforwardly modified to derive private algorithms with similar guarantees in this reverse scenario, where the source domain is private, and the target domain is public.

Handling distinct feature spaces. Our current formulation assumes the same input space for the source and target distributions. For adaptation problems with distinct input feature spaces, two approaches can be taken to extend our results: (1) If the mapping Ψ between the spaces is known, our theory can be readily extended to accommodate this situation. In this case, the adaptation algorithm can be adjusted by incorporating the mapping Ψ in the learning process; we consider $(h \circ \Psi)(x)$ for source domain instances; (2) If the mapping Ψ is unknown, it needs to be learned simultaneously with h. In this scenario, we consider $(h \circ \Psi)(x)$ for source domain instances, with Ψ being an integral part of the learning process. We leave a detailed analysis, including a needed extension of the generalization bound of Theorem B.1, to future work. The case of distinct output spaces is similar.

Unsupervised adaptation. Our study primarily focused on supervised adaptation. However, the theoretical framework and algorithms presented can be generalized to unsupervised or weakly supervised settings by leveraging the notion of *unlabeled discrepancy* (Mansour et al., 2009; Awasthi et al., 2024). Specifically, we can leverage the established bounds on labeled discrepancy in terms of unlabeled discrepancy (Awasthi et al., 2024) to extend our analysis and reparameterization techniques. By integrating our results with Theorem 6 and Corollary 7 from Awasthi, Cortes, and Mohri (2024)[Section 5], we can derive a differentially private adaptation algorithm for unsupervised or weakly supervised scenarios. The resulting algorithm can be viewed as a differentially private counterpart of the BEST-DA algorithm proposed in (Awasthi et al., 2024) [2024, Section 5]. We will present a more extensive analysis of this algorithm in future work.

8. Conclusion

We presented two (ε, δ)-differentially private algorithms for supervised adaptation based on strong theoretical learning guarantees. Our experimental results suggest that these algorithms can be effective in applications and scenarios where domain adaptation can be successful. Our work includes several key contributions. We established strong theoretical foundations for our algorithms, ensuring both privacy and learning guarantees. We introduced a new reparameterization technique, which we believe holds broader applicability in analyzing other related problems within private machine learning. Lastly, we validated the practical effectiveness of our algorithms through comprehensive experiments on real-world datasets, showcasing their potential in privacy-preserving domain adaptation. We believe that these algorithms represent a step towards practical, privacy-preserving domain adaptation, enabling us to leverage sensitive data for real-world applications, while protecting individual privacy.

Acknowledgements

Raef Bassily's research is supported by NSF CAREER Award 2144532 and NSF Award 2112471. We thank our colleague, Pranjal Awasthi, for early discussions about this work.

Impact Statement

Differentially private adaptation enables models to learn from public datasets while safeguarding privacy in private target domains. In healthcare, it can unlock diagnostic and personalized treatments, while in finance and law enforcement, it can promote fairness by mitigating biases. This technique's potential is significant, but addressing risks and ensuring ethical use is crucial. Through responsible implementation and robust privacy guarantees, differential privacy adaptation can revolutionize data-driven societal interactions, balancing privacy and utility to enhance healthcare, reduce bias, and improve societal well-being.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Alon, N., Bassily, R., and Moran, S. Limits of private learning with access to public data. *NeuRIPS 2019, also available at arXiv:1910.11519 [cs.LG]*, 2019.

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. Lower bounds for non-convex stochastic optimization, 2019.

Awasthi, P., Cortes, C., and Mohri, M. Best-effort adaptation. *Ann. Math. Artif. Intell.*, 92(2):393–438, 2024.

Balle, B. and Wang, Y.-X. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pp. 394–403. PMLR, 2018.

Bassily, R., Smith, A., and Thakurta, A. 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.

Bassily, R., Thakurta, A., and Thakkar, O. Model-agnostic private learning. In *Advances in Neural Information Processing Systems 31*, pp. 7102–7112. Curran Associates, Inc., 2018.

Bassily, R., Feldman, V., Talwar, K., and Guha Thakurta, A. Private stochastic convex optimization with optimal rates. *Advances in neural information processing systems*, 32, 2019.

Bassily, R., Cheu, A., Moran, S., Nikolov, A., Ullman, J., and Wu, S. Private query release assisted by public data. In *International Conference on Machine Learning*, pp. 695–703. PMLR, 2020.

Bassily, R., Mohri, M., and Suresh, A. T. Private domain adaptation from a public source. *CoRR*, abs/2208.06135, 2022.

Beck, A. *First-order methods in optimization*. SIAM, 2017.

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

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

Ben-David, S., Bie, A., Canonne, C. L., Kamath, G., and Singhal, V. Private distribution learning with public data: The view from sample compression. *CoRR*, abs/2308.06239, 2023.

Bie, A., Kamath, G., and Singhal, V. Private estimation with public data. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A. (eds.), *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*, 2022.

Blitzer, J., Dredze, M., and Pereira, F. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of ACL*, pp. 440–447, 2007.

Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Wortman, J. Learning bounds for domain adaptation. In *Proceedings of NIPS*, pp. 129–136, 2008.

Boyd, S. P. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, 2014.

Bun, M. and Steinke, T. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pp. 635–658. Springer, also, available at arXiv:1605.02065, 2016.

Chaudhuri, K. and Hsu, D. Sample complexity bounds for differentially private learning. In *Proceedings of the 24th Annual Conference on Learning Theory*, pp. 155–186, 2011.

Cortes, C. and Mohri, M. Domain adaptation in regression. In *Proceedings of ALT*, pp. 308–323, 2011.

Cortes, C. and Mohri, M. Domain adaptation and sample bias correction theory and algorithm for regression. *Theor. Comput. Sci.*, 519:103–126, 2014.

Cortes, C., Mansour, Y., and Mohri, M. Learning bounds for importance weighting. In *Proceedings of NIPS*, pp. 442–450. Curran Associates, Inc., 2010.

Cortes, C., Mohri, M., and Muñoz Medina, A. Adaptation based on generalized discrepancy. *J. Mach. Learn. Res.*, 20:1:1–1:30, 2019.

de Mathelin, A., Mougeot, M., and Vayatis, N. Discrepancy-based active learning for domain adaptation. *CoRR*, abs/2103.03757, 2021.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Li, F.-F. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

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

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science*, 9(3–4):211–407, 2014a.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Found. Trends Theor. Comput. Sci.*, 9(3-4):211–407, 2014b.

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

Dwork, C., McSherry, F., Nissim, K., and Smith, A. D. Calibrating noise to sensitivity in private data analysis. In *Proceedings of Theory of Cryptography Conference TCC*, volume 3876 of *Lecture Notes in Computer Science*, pp. 265–284. Springer, 2006b.

Fernandes, K. A proactive intelligent decision support system for predicting the popularity of online news. In *Springer Science and Business Media LLC'<*, 08 2015.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A PAC-bayesian approach for domain adaptation with specialization to linear classifiers. In *Proceedings of ICML*, volume 28 of *JMLR Workshop and Conference Proceedings*, pp. 738–746. JMLR.org, 2013.

Ghadimi, S., Lan, G., and Zhang, H. Mini-batch stochastic approximation methods for nonconvex stochastic composite optimization. *Mathematical Programming*, 155(1): 267–305, 2016.

Graf, F., Kriegel, H.-P., Schubert, M., Poelsterl, S., and Cavallaro, A. Relative location of CT slices on axial axis. UCI Machine Learning Repository, 2011. DOI: https://doi.org/10.24432/C5CP6G.

Hanneke, S. and Kpotufe, S. On the value of target data in transfer learning. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pp. 9867–9877, 2019.

Haslett, J. and Raftery, A. E. Space-time modeling with long-memory dependence: assessing ireland's wind-power resource. technical report. *Journal of the Royal Statistical Society*, 38(1), 1989.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Huang, J., Smola, A. J., Gretton, A., Borgwardt, K. M., and Schölkopf, B. Correcting sample selection bias by unlabeled data. In *NIPS 2006*, volume 19, pp. 601–608, 2006.

Ikonomovska, E. Airline dataset. Online, 2009. URL http://kt.ijs.si/elena_ikonomovska/data.html.

Jin, K., Cheng, X., Yang, J., and Shen, K. Differentially private correlation alignment for domain adaptation. In *IJCAI*, volume 21, pp. 3649–3655, 2021.

Kifer, D., Ben-David, S., and Gehrke, J. Detecting change in data streams. In *Proceedings of VLDB*, pp. 180–191. Morgan Kaufmann, 2004.

Krizhevsky, A. Learning multiple layers of features from tiny images. Technical report, Toronto University, 2009.

Ledoux, M. and Talagrand, M. *Probability in Banach Spaces: Isoperimetry and Processes*. Springer, New York, 1991.

Li, Q. Literature survey: domain adaptation algorithms for natural language processing. *Department of Computer Science The Graduate Center, The City University of New York*, pp. 8–10, 2012.

Li, Z. and Li, J. A simple proximal stochastic gradient method for nonsmooth nonconvex optimization. *Advances in neural information processing systems*, 31, 2018.

Lu, N., Zhang, T., Fang, T., Teshima, T., and Sugiyama, M. Rethinking importance weighting for transfer learning. *CoRR*, abs/2112.10157, 2021.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation: Learning bounds and algorithms. In *Proceedings of COLT*, 2009.

Mohri, M. and Muñoz Medina, A. New analysis and algorithm for learning with drifting distributions. In *Proceedings of ALT*, volume 7568 of *Lecture Notes in Computer Science*, pp. 124–138. Springer, 2012.

Mohri, M., Rostamizadeh, A., and Talwalkar, A. *Foundations of Machine Learning*. MIT Press, second edition, 2018.

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

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading digits in natural images with unsupervised feature learning. In *Advances in Neural Information Processing Systems*, 2011.

Pan, S. J. and Yang, Q. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10): 1345–1359, 2009.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PMLR, 2021.

Rodriguez-Lujan, I., Fonollosa, J., Vergara, A., Homer, M., and Huerta, R. On the calibration of sensor arrays for pattern recognition using the minimal number of experiments. *Chemometrics and Intelligent Laboratory Systems*, 130:123–134, 2014. ISSN 0169-7439.

Shamir, O. Can we find near-approximately-stationary points of nonsmooth nonconvex functions? *arXiv preprint arXiv:2002.11962*, 2020.

Sriperumbudur, B. K., Torres, D. A., and Lanckriet, G. R. G. Sparse eigen methods by D.C. programming. In *ICML*, pp. 831–838, 2007.

Steinke, T. and Ullman, J. Between pure and approximate differential privacy. *arXiv preprint arXiv:1501.06095*, 2015.

Sugiyama, M., Krauledat, M., and Müller, K. Covariate shift adaptation by importance weighted cross validation. *J. Mach. Learn. Res.*, 8:985–1005, 2007a.

Sugiyama, M., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. Direct importance estimation with model selection and its application to covariate shift adaptation. In *Proceedings of NIPS*, pp. 1433–1440. Curran Associates, Inc., 2007b.

Tao, P. D. and An, L. T. H. A DC optimization algorithm for solving the trust-region subproblem. *SIAM Journal on Optimization*, 8(2):476–505, 1998.

Tramèr, F., Kamath, G., and Carlini, N. Considerations for differentially private learning with large-scale public pretraining. *CoRR*, abs/2212.06470, 2022.

Vergara, A., Vembu, S., Ayhan, T., Ryan, M. A., Homer, M. L., and Huerta, R. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166-167:320–329, 2012. ISSN 0925-4005.

Wang, B., Mendez, J. A., Cai, M., and Eaton, E. Transfer learning via minimizing the performance gap between domains. In *Proceedingz of NeurIPS*, pp. 10644–10654, 2019.

Wang, D. and Xu, J. Differentially private empirical risk minimization with smooth non-convex loss functions: A non-stationary view. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1182–1189, 2019.

Wang, D., Ye, M., and Xu, J. Differentially private empirical risk minimization revisited: Faster and more general. *Advances in Neural Information Processing Systems*, 30, 2017.

Wang, M. and Deng, W. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

Wang, Q., Li, Z., Zou, Q., Zhao, L., and Wang, S. Deep domain adaptation with differential privacy. *IEEE Transactions on Information Forensics and Security*, 15:3093–3106, 2020.

Yuille, A. L. and Rangarajan, A. The concave-convex procedure. *Neural Computation*, 15(4):915–936, 2003.

Zhang, T., Yamane, I., Lu, N., and Sugiyama, M. A one-step approach to covariate shift adaptation. In *Proceedings of ACML*, volume 129 of *Proceedings of Machine Learning Research*, pp. 65–80. PMLR, 2020a.

Zhang, Y., Liu, T., Long, M., and Jordan, M. I. Bridging theory and algorithm for domain adaptation. In *Proceedings of ICML*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7404–7413. PMLR, 2019.

Zhang, Y., Long, M., Wang, J., and Jordan, M. I. On localized discrepancy for domain adaptation. *CoRR*, abs/2008.06242, 2020b.

Contents of Appendix

A Related work **15**

B General analysis of supervised adaptation **15**

B.1 General learning bound 16

B.2 Optimization problem 17

C Proofs of Section 4 **17**

C.1 Proof of Theorem 4.1 17

C.2 Proof of Lemma 4.2 18

C.3 Proof of Theorem 4.3 19

D Proofs of Section 5 **21**

D.1 Proof of Lemma 5.1 21

D.2 Formal description of Algorithm NCnvxAdap of Section 5 23

D.3 Proof of Theorem 5.2 23

E Additional experimental results **26**

E.1 Convex setting 26

E.2 Non-convex setting 27

A. Related work

There is a very broad literature on domain adaptation that we cannot survey in detail within this limited space. Thus, we refer the reader to surveys such as (Pan & Yang, 2009; Wang & Deng, 2018; Li, 2012) for a relatively comprehensive overview and briefly discuss approaches that are the most relevant to our study.

Our analysis admits a strong theoretical component since we seek a differentially private algorithm with theoretical learning and privacy guarantees. We benefit from several past publications already referenced that have given a theoretical analysis of adaptation using the notion of discrepancy. There are several other related publications using the notion of discrepancy for a PAC-Bayesian analysis (Germain et al., 2013) or active learning (de Mathelin et al., 2021). There are also other interesting theoretical analyses of adaptation such as (Hanneke & Kpotufe, 2019), which deals with the notions of super transfer or localization; these notions admit some connections with that of (local) discrepancy (Cortes et al., 2019; Zhang et al., 2020b).

In the privacy literature, several interesting algorithms have been given with formal differentially private learning guarantees, assuming access to public data (Chaudhuri & Hsu, 2011; Beimel et al., 2013; Bassily et al., 2018; Alon et al., 2019; Nandi & Bassily, 2020; Bassily et al., 2020). But these results cannot be used in the adaptation scenario we consider since they assume that the source and target domains coincide. A differentially private correlation alignment approach for domain adaptation was given by Jin et al. (2021) for a distinct scenario where both source and target data are private. More recently, Wang et al. (2020) described algorithms for deep domain adaptation for classification, but the authors do not provide theoretical guarantees for these algorithms.

The problem of private density estimation using a small amounto public data has been studied in several recent publications. Bie et al. (2022) studied the problem of estimating a d-dimensional Gaussian distribution, under the assumption of access to a Gaussian that may have vanishing similarity in total variation distance with the underlying Gaussian of the private data. Ben-David et al. (2023) studied the problem of private distribution learning with access to public data. They related private density estimation to sample compression schemes for distributions. They approximately recovered previous results on Gaussians, and presented other results such as sample complexity upper bounds for arbitrary k-mixtures of Gaussians. Tramèr et al. (2022) presented a general discussion of the question of private learning with large-scale public pretraining.

As discussed in the main text, the most closely related work to ours is the recent study of Bassily et al. (2022), which considers a similar adaptation scenario with a public source domain and a private target domain and which also gives private algorithms with theoretical guarantees. However, that work can be distinguished from ours in several aspects. First, the authors consider a purely unsupervised adaptation scenario where no labeled sample is available from the target domain, while we consider a supervised scenario. Our study and algorithms can be extended to the unsupervised or weakly supervised setting using the notion of *unlabeled discrepancy* (Mansour et al., 2009), by leveraging upper bounds on *labeled discrepancy* in terms of unlabeled discrepancy as in (Awasthi et al., 2024). Second, the learning guarantees of our private algorithms benefit from the recent optimization of Awasthi et al. (2024), which they show are theoretically stronger than those of the DM solution of Cortes & Mohri (2014) adopted by Bassily et al. (2022). Similarly, in our experiments, our convex optimization solution outperforms the DM algorithm. Note that the empirical study in (Bassily et al., 2022) is limited to a single specific artificial dataset, while we present empirical results with several non-artificial datasets. Third, our private adaptation algorithms include solutions both for regression and classification, while those of Bassily et al. (2022) are specifically given for regression with the squared loss.

B. General analysis of supervised adaptation

In this section, we describe the general learning bound of Awasthi et al. (2024), for which we give a self-contained and concise proof. This bound holds for any sample reweighting method in domain adaptation. This includes as special cases a number of methods presented for adaptation in the past, including KMM (Huang et al., 2006), KLIEP (Sugiyama et al., 2007b), importance weighting with bounded weights (Cortes et al., 2010), discrepancy minimization (Cortes & Mohri, 2014), gapBoost algorithm (Wang et al., 2019), and many others. Next, we discuss the implications of this bound and the related optimization problem.

B.1. General learning bound

The learning bound draws on a natural extension of the notion of Rademacher complexity to the weighted case, q-*weighted Rademacher complexity*, which is denoted by $\mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H})$ and defined by

$$\mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) = \mathop{\mathbb{E}}_{S^{\mathsf{Pub}}, S^{\mathsf{Priv}}, \boldsymbol{\sigma}}\left[\sup_{h \in \mathcal{H}} \sum_{i=1}^{m+n} \sigma_i \mathsf{q}_i \ell(h(x_i), y_i)\right], \tag{8}$$

where σ_is are independent random variables uniformly distributed over $\{-1, +1\}$. The bound holds uniformly over both the choice of a hypothesis h selected in $\mathcal{H}$ and that of a weight vector q in the open $\|\cdot\|_1$-ball of radius one centered in p^0, $\mathsf{B}_1(\mathsf{p}^0, 1) = \{\mathsf{q}\colon \|\mathsf{q} - \mathsf{p}^0\|_1 < 1\}$, where p^0 can be interpreted as a *reference* or *ideal* reweighting choice.

Theorem B.1. *For any $\eta > 0$, with probability at least $1 - \eta$ over the draw of a sample S^{Pub} of size m from $\mathcal{D}^{\mathsf{Pub}}$ and a sample S^{Priv} of size n from $\mathcal{D}^{\mathsf{Priv}}$, the following holds for all $h \in \mathcal{H}$ and $\mathsf{q} \in \mathsf{B}_1(\mathsf{p}^0, 1)$:*

$$\begin{aligned}\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) \le \sum_{i=1}^{m+n} \mathsf{q}_i \ell(h(x_i), y_i) + \overline{\mathsf{q}}^{\mathsf{Pub}} \mathrm{dis}\big(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}\big) + 2\mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) + 6B\|\mathsf{q} - \mathsf{p}^0\|_1 \\ + B\big[\|\mathsf{q}\|_2 + 2\|\mathsf{q} - \mathsf{p}^0\|_1\big]\left[\sqrt{\log\log_2 \tfrac{2}{1 - \|\mathsf{q} - \mathsf{p}^0\|_1}} + \sqrt{\tfrac{\log(2/\eta)}{2}}\right].\end{aligned}$$

Proof. The proof of theorem consists of first deriving a q-weighted Rademacher complexity bound for a fixed reweighting q, using the fact that the expectation of the empirical term is then

$$\big(\overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h)\big).$$

Next, the difference of $\mathsf{q}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h)$ and this term is analyzed in terms of the discrepancy term $\overline{\mathsf{q}}^{\mathsf{Pub}} \mathrm{dis}\big(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}\big)$ and then the bound is extended to hold uniformly over $\mathsf{B}_1(\mathsf{p}^0, 1)$, using a technique similar to that of deriving uniform margin bounds, see for example (Mohri et al., 2018)[Chapter 5]. We will use S to refer to the full sample: $S = (S^{\mathsf{Pub}}, S^{\mathsf{Priv}})$ and will use the shorthand $\mathcal{L}_S(\mathsf{q}, h) = \sum_{i=1}^{m+n} \mathsf{q}_i \ell(h(x_i), y_i)$ for the empirical term.

Fix $\mathsf{q} \in [0, 1]^{m+n}$. The expectation of $\mathcal{L}_S(\mathsf{q}, h)$ over the draw of $(S^{\mathsf{Pub}}, S^{\mathsf{Priv}}) \sim (\mathcal{D}^{\mathsf{Pub}})^m \times (\mathcal{D}^{\mathsf{Pub}})^n$ is then given by

$$\begin{aligned}\mathop{\mathbb{E}}_{S}[\mathcal{L}_S(\mathsf{q}, h)] &= \sum_{i=1}^{m} \mathsf{q}_i \mathop{\mathbb{E}}_{S^{\mathsf{Pub}}}[\ell(h(x_i), y_i)] + \sum_{i=m+1}^{m+n} \mathsf{q}_i \mathop{\mathbb{E}}_{S^{\mathsf{Priv}}}[\ell(h(x_i), y_i)] \\ &= \sum_{i=1}^{m} \mathsf{q}_i \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \sum_{i=m+1}^{m+n} \mathsf{q}_i \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) \\ &= \overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h).\end{aligned}$$

Consider $\Psi(S) = \sup_{h \in \mathcal{H}}\big\{\overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \mathcal{L}_S(\mathsf{q}, h)\big\}$. Changing point x_i to x_i' affects $\Psi(S)$ at most by $\mathsf{q}_i B$, since the loss is bounded by B. It is also not hard to see that the standard symmetrization argument (see (Mohri et al., 2018)) can be extended to the weighted case and that $\mathbb{E}_S[\Psi(S)] \le 2\mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H})$. Thus, by McDiarmid's inequality, for any $\eta > 0$, with probability at least $1 - \eta$, the following holds:

$$\overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \mathcal{L}_S(\mathsf{q}, h) \le 2\mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) + B\sqrt{\frac{\log \frac{1}{\eta}}{2m}}. \tag{9}$$

Now, we can also analyze the difference of $\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h)$ and $\overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h)$ as follows:

$$\begin{aligned}&\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \big(\overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) + \overline{\mathsf{q}}^{\mathsf{Priv}} \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h)\big) \\ &= (1 - \overline{\mathsf{q}}^{\mathsf{Priv}}) \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \overline{\mathsf{q}}^{\mathsf{Pub}} \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h) \\ &= (1 - \overline{\mathsf{q}}^{\mathsf{Priv}} - \overline{\mathsf{q}}^{\mathsf{Pub}}) \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) + \overline{\mathsf{q}}^{\mathsf{Pub}} (\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h)) \\ &= (\|\mathsf{p}^0\|_1 - \|\mathsf{q}\|_1) \mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) + \overline{\mathsf{q}}^{\mathsf{Pub}} (\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) - \mathcal{L}(\mathcal{D}^{\mathsf{Pub}}, h)) \\ &\le B\|\mathsf{p}^0 - \mathsf{q}\|_1 + \overline{\mathsf{q}}^{\mathsf{Pub}} \mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}).\end{aligned} \tag{10}$$

Combining (9) and (10) yields the following high-probability inequality for a fixed q:

$$\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) \le \mathcal{L}_S(\mathsf{q}, h) + 2B\|\mathsf{p}^0 - \mathsf{q}\|_1 + \overline{\mathsf{q}}^{\mathsf{Pub}}\mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}) + \mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) + B\sqrt{\frac{\log \frac{1}{\eta}}{2m}}. \tag{11}$$

Consider a sequence of weight vectors $\mathsf{q}_k \in [0,1]^{m+n}$ and a sequence of confidence weights $\eta_k = \frac{\eta}{2^{k+1}}$. Inequality (11), with q replaced by q^k and η replaced by η_k, holds for each $k \ge 0$, with probability $1 - \eta_k$. Thus, by the union bound, since $\sum_{k=0}^{+\infty} \frac{\eta}{2^{k+1}} = \eta$, with probability $1 - \eta$, it holds for all $k \ge 0$.

$$\mathcal{L}(\mathcal{D}^{\mathsf{Priv}}, h) \le \mathcal{L}_S(\mathsf{q}^k, h) + 2B\|\mathsf{p}^0 - \mathsf{q}^k\|_1 + \overline{\mathsf{q}}^{k,\mathsf{Pub}}\mathrm{dis}(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}) + \mathfrak{R}_{\mathsf{q}^k}(\ell \circ \mathcal{H}) + B\sqrt{\frac{\log \frac{1}{\eta}}{2m}}. \tag{12}$$

We can choose q^k such that $\|\mathsf{q}^k - \mathsf{p}^0\|_1 = 1 - \frac{1}{2^k}$. Then, for any $\mathsf{q} \in \mathsf{B}(\mathsf{p}^0, 1)$, there exists $k \ge 0$ such that $\|\mathsf{q}^k - \mathsf{p}^0\|_1 \le \|\mathsf{q} - \mathsf{p}^0\|_1 < \|\mathsf{q}^{k+1} - \mathsf{p}^0\|_1$ and thus such that

$$\sqrt{2\log(k+1)} = \sqrt{2\log\log_2 \frac{2}{1 - \|\mathsf{q}^k - \mathsf{p}^0\|_1}} \le \sqrt{2\log\log_2 \frac{2}{1 - \|\mathsf{q} - \mathsf{p}^0\|_1}}.$$

Furthermore, for that k, the following inequalities hold:

$$\begin{aligned}
\mathcal{L}_S(\mathsf{q}^k, h) &\le \sum_{i=1}^{m+n} \mathsf{q}_i \ell(h(x_i), y_i) + B\|\mathsf{q}^k - \mathsf{q}\|_1 \le \mathcal{L}_S(\mathsf{q}, h) + 2B\|\mathsf{p}^0 - \mathsf{q}\|_1 \\
\overline{\mathsf{q}}^{k,\mathsf{Pub}} &\le \overline{\mathsf{q}}^{\mathsf{Pub}} + \|\mathsf{q}^k - \mathsf{q}\|_1 \le \overline{\mathsf{q}}^{\mathsf{Pub}} + 2\|\mathsf{p}^0 - \mathsf{q}\|_1 \\
\mathfrak{R}_{\mathsf{q}^k}(\ell \circ \mathcal{H}) &\le \mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) + B\|\mathsf{q}^k - \mathsf{q}\|_1 \le \mathfrak{R}_{\mathsf{q}}(\ell \circ \mathcal{H}) + 2B\|\mathsf{q} - \mathsf{p}^0\|_1, \\
\|\mathsf{q}^k\|_2 &\le \|\mathsf{q}\|_2 + \|\mathsf{q}^k - \mathsf{q}\|_2 \le \|\mathsf{q}\|_2 + \|\mathsf{q}^k - \mathsf{q}\|_1 \le \|\mathsf{q}\|_2 + 2\|\mathsf{q} - \mathsf{p}^0\|_1.
\end{aligned}$$

Plugging in these inequalities in (12) completes the proof. □

B.2. Optimization problem

The bound suggests the following to achieve a good generalization error in adaptation: ensure a small q-empirical loss (first term), but not at the price of a too sparse weight vector, which would result in a larger $\|\mathsf{q}\|_2$ (fifth term); allocate a smaller total weight to public points when the discrepancy is larger (second term); limit the q-weighted complexity of the hypothesis set $\mathcal{H}$ combined with the loss function ℓ (third term); and ensure the closeness of q to the reference weight vector p^0 (fourth and fifth terms).

The joint optimization problem (3) is directly based on minimizing the right-hand side of the inequality over the choice of both $h \in \mathcal{H}$ and $\mathsf{q} \in \mathsf{B}_1(\mathsf{p}^0, 1)$. By McDiarmid's inequality, the discrepancy term $\mathrm{dis}\big(\mathcal{D}^{\mathsf{Priv}}, \mathcal{D}^{\mathsf{Pub}}\big)$ can be replaced by its estimate $\hat{d}$ (2) from finite sample modulo a term in $O(\sqrt{\frac{m+n}{mn}})$. Instead of the supremum over the full family $\mathcal{H}$, one can also use a local discrepancy (Cortes et al., 2019; de Mathelin et al., 2021; Zhang et al., 2019; 2020b) and restrict oneself to a ball around the empirical minimizer of the private loss of radius $O(1/\sqrt{n})$. Using Talagrand's inequality (Ledoux & Talagrand, 1991) and the straightforward observation that for any i, $x \mapsto \mathsf{q}_i x$ is $\|\mathsf{q}\|_\infty$-Lipschitz, the weighted Rademacher complexity bound can be upper bounded by $\|\mathsf{q}\|_\infty (m+n)\mathfrak{R}_{m+n}(\ell \circ \mathcal{H})$, where $\mathfrak{R}_{m+n}(\ell \circ \mathcal{H})$ is the standard (unweighted) Rademacher complexity of the family of loss functions over the hypothesis set $\mathcal{H}$. For uniform weights, this is an equality. Thus, using the upper bounds just discussed and replacing constants with hyperparameters, minimizing the right-hand side of the learning bounds of Theorem B.1 can be formulated as the joint optimization problem (3).

C. Proofs of Section 4

C.1. Proof of Theorem 4.1

Theorem 4.1. *Let $\mathsf{G}(w, \mathsf{q})$ denote the objective function of problem (5) and (w^*, q^*) its minimizer (the solution of (5)). Let $(\tilde{w}, \tilde{\mathsf{u}})$ be the minimizer of $\mathsf{F}(w, \mathsf{u})$ (the solution of (6)). Define $\tilde{\mathsf{q}} \in [0,1]^{m+n}$ as the weight vector obtained by applying the inverse transformation to $\tilde{\mathsf{u}}$: $\tilde{\mathsf{q}}_i \to \frac{1}{\tilde{\mathsf{u}}_i}, \forall i \in [m+n]$. Assume that there exists a universal constant $C \ge 1$ such that $\forall i \in [m+n], C\mathsf{q}^*_i \ge \mathsf{p}^0_i$. Then, then the following inequality holds:*

$$\mathsf{G}(\tilde{w}, \tilde{\mathsf{q}}) \le \mathsf{G}(w^*, \mathsf{q}^*) + O(\|\mathsf{p}^0 - \mathsf{q}^*\|_1).$$

Proof. We maintain identical hyperparameter settings for both the original objective (problem (5)) and the relaxed objective (problem (6)). Define

$$g(\mathsf{u}) \triangleq 1 - \sum_{i=1}^{m+n} \frac{1}{\mathsf{u}_i} \quad \text{and} \quad f(\mathsf{u}) \triangleq \left(\frac{\alpha}{m}\right)^2 \sum_{i=1}^{m} \mathsf{u}_i + \left(\frac{1-\alpha}{n}\right)^2 \sum_{i=m+1}^{m+n} \mathsf{u}_i - 1.$$

Here, $g(\mathsf{u})$ represents the non-convex (concave) term in the original objective after reparameterization ($\mathsf{q} \to \mathsf{u}$, as defined in Section 4), and $f(\mathsf{u})$ is its convex upper bound, which replaces $g(\mathsf{u})$ in the convex objective (6). We first bound the difference $f(\mathsf{u}) - g(\mathsf{u})$ in terms of the second-order term of the Taylor expansion of $g(\mathsf{u})$ around $\mathsf{u} = \left(\frac{1}{\mathsf{p}_1^0}, \ldots, \frac{1}{\mathsf{p}_{m+n}^0}\right)$, where $\mathsf{p}^0 = \left(\mathsf{p}_1^0, \ldots, \mathsf{p}_{m+n}^0\right)$ is the reference weight vector. Note that $f(\mathsf{u})$ is the first-order term of the Taylor expansion of $g(\mathsf{u})$ (and the zeroth-order term is zero), and hence, the difference $f(\mathsf{u}) - g(\mathsf{u})$ can be bounded in terms of the second-order term, namely, $f(\mathsf{u}) - g(\mathsf{u}) = O(\mathcal{E}_{\mathsf{u}})$, where $\mathcal{E}_{\mathsf{u}} = \sum_{i=1}^{m+n} \left(\mathsf{p}_i^0\right)^3 \left(\mathsf{u}_i - \frac{1}{\mathsf{p}_i^0}\right)^2$. By applying the inverse transformation $\mathsf{u}_i \to 1/\mathsf{q}_i, \forall i \in [m+n]$, we write the above error in terms of q: $\mathcal{E}_{\mathsf{q}} = \sum_{i=1}^{m+n} \mathsf{p}_i^0 \left(\frac{\mathsf{p}_i^0}{\mathsf{q}_i} - 1\right)^2$.

Recall that (w^*, q^*) is the solution of the original optimization problem (5). Note that in Section 4 we showed that (5) is equivalent to (4) under the specific form of p^0 we use. Next, we establish an upper bound for $\mathcal{E}_{\mathsf{q}^*}$ under the assumption that there is a universal constant $C \geq 1$ such that $\forall i \in [m+n], C\mathsf{q}_i^* \geq \mathsf{p}_i^0$. Note that, by default, for all $i \in [m+n]$, we must have $\mathsf{q}_i^* \leq \mathsf{p}_i^0$ by the constraints in the original problem (5).

Now, observe that the following holds:

$$\begin{aligned}
\mathcal{E}_{\mathsf{q}^*} &= \sum_{i=1}^{m+n} \mathsf{p}_i^0 \left[\frac{\mathsf{p}_i^0}{\mathsf{q}_i^*} - 1\right]^2 \\
&\leq (C-1) \sum_{i=1}^{m+n} \mathsf{p}_i^0 \left[\frac{\mathsf{p}_i^0}{\mathsf{q}_i^*} - 1\right] \\
&= (C-1) \sum_{i=1}^{m+n} \mathsf{p}_i^0 \left[\frac{\mathsf{p}_i^0 - \mathsf{q}_i^*}{\mathsf{q}_i^*}\right] \\
&\leq C(C-1) \sum_{i=1}^{m+n} \left[\mathsf{p}_i^0 - \mathsf{q}_i^*\right] \\
&= C(C-1) \left\|\mathsf{p}^0 - \mathsf{q}^*\right\|_1,
\end{aligned}$$

where the second inequality follows from $\frac{\mathsf{p}_i^0}{\mathsf{q}_i^*} - 1 \leq C - 1$ by assumption, the fourth inequality follows from $\mathsf{q}_i^* \geq \frac{\mathsf{p}_i^0}{C}$ again by the same assumption, and the last inequality follows from the fact that $\mathsf{q}_i^* \leq \mathsf{p}_i^0, \forall i \in [m+n]$.

We now present the final step of the proof. Recall that $\mathsf{G}(w, \mathsf{q})$ denotes the original objective (5) and $\mathsf{F}(w, \mathsf{u})$ denotes the convex objective (6). Recall also that $(\tilde{w}, \tilde{\mathsf{u}})$ denotes the solution of (6) (i.e., the minimizer of F) and $\tilde{\mathsf{q}}$ denotes the weight vector corresponding to $\tilde{\mathsf{u}}$ after applying the inverse transformation. Let u^* denote the parameterization of q^*, that is, $\mathsf{u}_i^* = \frac{1}{\mathsf{q}_i^*}, \forall i \in [m+n]$. Observe that

$$\mathsf{G}(\tilde{w}, \tilde{\mathsf{q}}) \leq \mathsf{F}(\tilde{w}, \tilde{\mathsf{u}}) \leq \mathsf{F}(w^*, \mathsf{u}^*) \leq \mathsf{G}(w^*, \mathsf{q}^*) + O(\mathcal{E}_{\mathsf{q}^*}) = \mathsf{G}(w^*, \mathsf{q}^*) + O\left(\left\|\mathsf{p}^0 - \mathsf{q}^*\right\|_1\right),$$

where the first inequality follows because $\mathsf{F}(w, \mathsf{u})$ uniformly upper bounds $\mathsf{G}(w, \mathsf{q})$ (when u is the parameterization of q), the second inequality follows because $(\tilde{w}, \tilde{\mathsf{u}})$ is the minimizer of F, and the third inequality follows from $f(\mathsf{u}^*) - g(\mathsf{u}^*) = O(\mathcal{E}_{\mathsf{q}^*})$. □

C.2. Proof of Lemma 4.2

Lemma C.1. *The following properties hold for the objective function* F.

(i) The following upper bounds hold for the gradients, for all $(w, \mathsf{u}) \in \mathcal{W} \times \mathcal{U}$*:* $\|\nabla_w \mathsf{F}(w, \mathsf{u})\|_2 \leq G$*,* $\|\nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w, \mathsf{u})\|_2 \leq \frac{\alpha^2(B+\bar{B})}{m^{3/2}}$*, and* $\|\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w, \mathsf{u})\|_2 \leq \frac{(1-\alpha)^2 \bar{B}}{n^{3/2}}$.

(ii) The ℓ_2*-sensitivity of* $\nabla_w \widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$ *with respect to changing one private data point is at most* $\frac{2(1-\alpha)G}{n}$*;*

(iii) The ℓ_2*-sensitivity of* $\nabla_{\mathsf{u}^{\mathsf{Priv}}} \widehat{L}^{\mathsf{Priv}}(w, \mathsf{u}^{\mathsf{Priv}})$ *with respect to changing one private data point is at most* $\frac{(1-\alpha)^2 B}{n^2}$.

Proof. First observe that the following inequalities hold:

$$\|\nabla_w \mathsf{F}(w,\mathsf{u})\|_2 \le \sum_{i=1}^{m+n} \frac{\|\nabla_w \ell_{\mathsf{sq}}(w,(x_i,y_i))\|_2}{\mathsf{u}_i} \le G \sum_{i=1}^{n+m} \frac{1}{\mathsf{u}_i} \le G,$$

where the second inequality follows from the G-Lipschitzness of the loss, and the third from the constraints on u: $\frac{1}{\mathsf{u}_i} \le \frac{\alpha}{m}, \forall i \in [m]$ and $\frac{1}{\mathsf{u}_i} \le \frac{1-\alpha}{n}, \forall i \in [m+1, m+n]$.

Next, note that we have

$$\begin{aligned}
\left|\frac{\partial \mathsf{F}(w,\mathsf{u})}{\partial \mathsf{u}_i^{\mathsf{Pub}}}\right| &\le \frac{2B}{\left(\mathsf{u}_i^{\mathsf{Pub}}\right)^2} + \kappa_1 \frac{\alpha^2}{m} + \kappa_2 \frac{\frac{1}{(\mathsf{u}_i^{\mathsf{Pub}})^3}}{\sqrt{\sum_{i=1}^{m+n} \frac{1}{(\mathsf{u}_i^{\mathsf{Pub}})^2}}} + \kappa_\infty \frac{1\left(i \in \operatorname{argmin}_{j\in[m+n]} \mathsf{u}_j\right)}{\left(\mathsf{u}_i^{\mathsf{Pub}}\right)^2} \\
&\le \frac{2B}{\left(\mathsf{u}_i^{\mathsf{Pub}}\right)^2} + \kappa_1 \frac{\alpha^2}{m} + \frac{\kappa_2}{\left(\mathsf{u}_i^{\mathsf{Pub}}\right)^2} + \frac{\kappa_\infty}{\left(\mathsf{u}_i^{\mathsf{Pub}}\right)^2} \\
&\le \frac{\alpha^2}{m^2}(2B + \kappa_1 + \kappa_2 + \kappa_\infty),
\end{aligned}$$

where the first inequality follows from the fact that the loss is uniformly bounded by B, and hence $\ell_{\mathsf{sq}}(w,(x_i,y_i)) + \hat{d}_{\mathrm{DP}}\, 1_{i\le m} \le 2B$. The remaining steps follow straightforwardly from the constraints on $\mathsf{u}^{\mathsf{Pub}}$. Thus, we have $\|\nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w,\mathsf{u}^{\mathsf{Pub}})\|_2 \le \frac{\alpha^2}{m^{3/2}}(2B + \kappa_1 + \kappa_2 + \kappa_\infty)$.

Similarly, we have $\left|\frac{\partial \mathsf{F}(w,\mathsf{u})}{\partial \mathsf{u}_i^{\mathsf{Priv}}}\right| \le \frac{(1-\alpha)^2}{n^2}(B + \kappa_1 + \kappa_2 + \kappa_\infty)$ and thus

$$\|\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w,\mathsf{u}^{\mathsf{Priv}})\|_2 \le \frac{(1-\alpha)^2}{n^{3/2}}(B + \kappa_1 + \kappa_2 + \kappa_\infty).$$

This proves the first item of the lemma.

Second, we bound the ℓ_2-sensitivity of $\nabla_w \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}})$ and $\nabla_{\mathsf{u}^{\mathsf{Priv}}} \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}})$. Consider any pair of neighboring private datasets S^{Priv} and $\bar{S}^{\mathsf{Priv}}$. Let $(x_j^{\mathsf{Priv}}, y_j^{\mathsf{Priv}})$ and $(\bar{x}_j^{\mathsf{Priv}}, \bar{y}_j^{\mathsf{Priv}})$ be the data points by which the two datasets differ. To emphasize the dependence on the dataset, we will denote $\nabla_w \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}})$ with respect to dataset S as $\nabla_w \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}};S)$. We can write:

$$\begin{aligned}
&\left\|\nabla_w \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}};S^{\mathsf{Priv}}) - \nabla_w \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}};\bar{S}^{\mathsf{Priv}})\right\|_2 \\
&\quad = \frac{\left\|\nabla_w \ell_{\mathsf{sq}}(w,(x_j^{\mathsf{Priv}},y_j^{\mathsf{Priv}})) - \nabla_w \ell_{\mathsf{sq}}(w,(\bar{x}_j^{\mathsf{Priv}},\bar{y}_j^{\mathsf{Priv}}))\right\|_2}{\mathsf{u}_j^{\mathsf{Priv}}} \\
&\quad \le \frac{2G(1-\alpha)}{n}.
\end{aligned}$$

Similarly, we can write:

$$\begin{aligned}
&\left\|\nabla_{\mathsf{u}^{\mathsf{Priv}}} \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}};S^{\mathsf{Priv}}) - \nabla_{\mathsf{u}^{\mathsf{Priv}}} \widehat{L}^{\mathsf{Priv}}(w,\mathsf{u}^{\mathsf{Priv}};\bar{S}^{\mathsf{Priv}})\right\|_2 \\
&\quad = \left|\frac{\ell_{\mathsf{sq}}(w,(x_j^{\mathsf{Priv}},y_j^{\mathsf{Priv}})) - \ell_{\mathsf{sq}}(w,(\bar{x}_j^{\mathsf{Priv}},\bar{y}_j^{\mathsf{Priv}}))}{\left(\mathsf{u}_j^{\mathsf{Priv}}\right)^2}\right| \\
&\quad \le \frac{B(1-\alpha)^2}{n^2}.
\end{aligned}$$

This completes the proof. □

C.3. Proof of Theorem 4.3

Theorem 4.3. *For any $\delta > 0$ and $0 < \varepsilon \le 8\log(1/\delta)$, Algorithm 1 is (ε,δ)-differentially private. Furthermore, let (w^*,u^*) be a minimizer of $\mathsf{F}(w,\mathsf{u})$, then, the expected optimization error of the solution $(\bar{w},\bar{\mathsf{u}})$ returned by Algorithm 1 is bounded*

as follows:

$$\begin{aligned}&\mathsf{F}(\bar{w},\bar{\mathsf{u}}) - \mathsf{F}(w^*,\mathsf{u}^*)\\ &\le O\left(\frac{G\Lambda\sqrt{d\log\frac{1}{\delta}}}{n\varepsilon} + \frac{B\max\left(1,\|\mathsf{u}_0-\mathsf{u}^*\|_2^2\right)\sqrt{\log\frac{1}{\delta}}}{n^{3/2}\varepsilon}\right),\end{aligned}$$

for $T \ge \max\left(1, \frac{n^2\varepsilon^2}{d(1-\alpha)^2\log(\frac{1}{\delta})}, \frac{\bar{B}^2\varepsilon^2}{B^2\log(\frac{1}{\delta})}, \frac{\varepsilon^2\bar{B}^2n^3}{\log(\frac{1}{\delta})B^2m^3}\right)$.

We will show more precisely the following inequality:

$$\begin{aligned}&\mathbb{E}\left[\mathsf{F}(\bar{w},\bar{\mathsf{u}}) - \mathsf{F}(w^*,\mathsf{u}^*)\right]\\ &\le \Lambda G\sqrt{\frac{1}{T} + \frac{32(1-\alpha)^2 d\log(\frac{2}{\delta})}{n^2\varepsilon^2}}\\ &\quad + \frac{(1-\alpha)^2\left(U^{\mathsf{Priv}}+1\right)}{2n^{3/2}}\sqrt{\frac{\bar{B}^2}{T} + \frac{8B^2\log(\frac{2}{\delta})}{\varepsilon^2}}\\ &\quad + \frac{\alpha^2\left(U^{\mathsf{Pub}}+1\right)(B+\bar{B})}{2m^{3/2}\sqrt{T}},\end{aligned}$$

where $\mathsf{u}^* = \left(\mathsf{u}^{\mathsf{Pub}*}, \mathsf{u}^{\mathsf{Priv}*}\right)$, $U^{\mathsf{Priv}} \triangleq \|\mathsf{u}_0^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\|_2^2$, and $U^{\mathsf{Pub}} \triangleq \|\mathsf{u}_0^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\|_2^2$.

Proof. First, we show the privacy guarantee. At a high level, Algorithm 1 can be viewed as T-fold adaptive composition of the Gaussian mechanism, and hence, the privacy guarantee can be shown using the properties of the Gaussian mechanism and, particularly, how it composes adaptively. One way to show that is to resort to the privacy-loss random variable interpretation of (ε,δ)-differential privacy (Dwork & Roth, 2014b); namely, one can use the fact that privacy loss random variable of the Gaussian mechanism is also Gaussian (Balle & Wang, 2018) and the fact that the privacy loss of the T-fold adaptive composition is a sum of T such variables (and hence, is also Gaussian) to prove the desired privacy guarantee. A more direct approach for the proof is to use the notion of zero-Concentrated Differential Privacy (zCDP) (Bun & Steinke, 2016) and then transforming the final guarantee to (ε,δ)-differential privacy. Note that for any iteration $t \in [T]$ in Algorithm 1, the only quantities that depend on the private dataset are the gradient components $\nabla_w\mathsf{F}(w_t,\mathsf{u}_t)$ and $\nabla_{\mathsf{u}^{\mathsf{Priv}}}\mathsf{F}(w_t,\mathsf{u}_t)$. From the guarantees on the ℓ_2-sensitivity of these gradient components given by parts 2 and 3 of Lemma 4.2 and by the zCDP properties of the Gaussian mechanism (Bun & Steinke, 2016, Lemma 2.5), each iteration $t \in [T]$ of Algorithm 1 is $\frac{\varepsilon^2}{16T\log(1/\delta)}$-zCDP. Hence, by the adaptive composition of zCDP (Bun & Steinke, 2016, Lemma 2.3) over the T iterations of the algorithm, we get that Algorithm 1 is $\frac{\varepsilon^2}{16\log(1/\delta)}$-zCDP. By transforming this guarantee into an approximate differential privacy guarantee (Bun & Steinke, 2016, Lemma 3.5) and using the fact that $\varepsilon \le 8\log(1/\delta)$, we get that Algorithm 1 is (ε,δ)-differentially private.

Next, we prove the bound on the optimization error. The proof involves some tweaks of the the standard analysis of the (stochastic) projected gradient descent algorithm for convex objectives. In particular, our proof entails decomposing each gradient $\nabla\mathsf{F}$ into its three components $\nabla_w\mathsf{F}$, $\nabla_{\mathsf{u}^{\mathsf{Pub}}}\mathsf{F}$, and $\nabla\mathsf{F}_{\mathsf{u}^{\mathsf{Priv}}}$ to allow for introducing a different step size for updating each of w, $\mathsf{u}^{\mathsf{Pub}}$ and $\mathsf{u}^{\mathsf{Priv}}$. By a standard argument, we have

$$\begin{aligned}&\mathbb{E}\left[\|w_{t+1}-w^*\|_2^2\right]\\ &\le \mathbb{E}\left[\|w_t-w^*\|_2^2\right] - 2\eta_w\mathbb{E}\left[\langle\nabla_w\mathsf{F}(w_t,\mathsf{u}_t)+\mathbf{z}_t, w_t-w^*\rangle\right] + \eta_w^2\mathbb{E}\left[\|\nabla_w\mathsf{F}(w_t,\mathsf{u}_t)+\mathbf{z}_t\|_2^2\right]\\ &= \mathbb{E}\left[\|w_t-w^*\|_2^2\right] - 2\eta_w\mathbb{E}\left[\langle\nabla_w\mathsf{F}(w_t,\mathsf{u}_t), w_t-w^*\rangle\right] + \eta_w^2\mathbb{E}\left[\|\nabla_w\mathsf{F}(w_t,\mathsf{u}_t)+\mathbf{z}_t\|_2^2\right]\\ &\le \mathbb{E}\left[\|w_t-w^*\|_2^2\right] - 2\eta_w\mathbb{E}\left[\langle\nabla_w\mathsf{F}(w_t,\mathsf{u}_t), w_t-w^*\rangle\right] + \eta_w^2\left(G^2+\sigma_1^2 d\right),\end{aligned}$$

where the second step follows from the linearity of expectation and the fact that $\mathbf{z}_t$ is independent of $((w_0,\mathsf{u}_0),\ldots,(w_t,\mathsf{u}_t))$ and that $\mathbf{z}_t$ has zero mean, and the last step follows from the fact that $\|\nabla_w\mathsf{F}(w_t,\mathsf{u}_t)\|_2^2 \le G^2$ proved in Lemma 4.2 and the fact that $\|\mathbf{z}_t\|_2^2 = \sigma_1^2 d$. Hence,

$$\mathbb{E}\left[\langle\nabla_w\mathsf{F}(w_t,\mathsf{u}_t), w_t-w^*\rangle\right] \le \frac{\mathbb{E}\left[\|w_t-w^*\|_2^2\right] - \mathbb{E}\left[\|w_{t+1}-w^*\|_2^2\right]}{2\eta_w} + \frac{\eta_w}{2}\left(G^2+\sigma_1^2 d\right) \tag{13}$$

By a similar argument for $\|\mathsf{u}_{t+1}^{\mathsf{Pub}} - \mathsf{u}_t^{\mathsf{Pub}}\|_2^2$ and $\|\mathsf{u}_{t+1}^{\mathsf{Priv}} - \mathsf{u}_t^{\mathsf{Priv}}\|_2^2$ and using Lemma 4.2, we get

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w_t, \mathsf{u}_t), \mathsf{u}_t^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*} \rangle\right] \\
&\quad \leq \frac{\mathbb{E}\left[\|\mathsf{u}_t^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\|_2^2\right] - \mathbb{E}\left[\|\mathsf{u}_{t+1}^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\|_2^2\right]}{2\eta_{\mathsf{u}^{\mathsf{Pub}}}} + \frac{\eta_{\mathsf{u}^{\mathsf{Pub}}}}{2}\mathbb{E}\left[\|\nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w_t, \mathsf{u}_t)\|_2^2\right] \\
&\quad \leq \frac{\mathbb{E}\left[\|\mathsf{u}_t^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\|_2^2\right] - \mathbb{E}\left[\|\mathsf{u}_{t+1}^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\|_2^2\right]}{2\eta_{\mathsf{u}^{\mathsf{Pub}}}} + \frac{\eta_{\mathsf{u}^{\mathsf{Pub}}}}{2} \cdot \frac{\alpha^4(\bar{B} + B)^2}{m^3} \qquad (14)
\end{aligned}$$

$$\begin{aligned}
&\mathbb{E}\left[\langle \nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w_t, \mathsf{u}_t), \mathsf{u}_t^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*} \rangle\right] \\
&\quad \leq \frac{\mathbb{E}\left[\|\mathsf{u}_t^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\|_2^2\right] - \mathbb{E}\left[\|\mathsf{u}_{t+1}^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\|_2^2\right]}{2\eta_{\mathsf{u}^{\mathsf{Priv}}}} + \frac{\eta_{\mathsf{u}^{\mathsf{Priv}}}}{2}\mathbb{E}\left[\|\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w_t, \mathsf{u}_t) + \mathbf{z}_t'\|_2^2\right] \\
&\quad \leq \frac{\mathbb{E}\left[\|\mathsf{u}_t^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\|_2^2\right] - \mathbb{E}\left[\|\mathsf{u}_{t+1}^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\|_2^2\right]}{2\eta_{\mathsf{u}^{\mathsf{Priv}}}} + \frac{\eta_{\mathsf{u}^{\mathsf{Priv}}}}{2}\left(\frac{(1-\alpha)^4\bar{B}^2}{n^3} + \sigma_2^2 n\right) \qquad (15)
\end{aligned}$$

By convexity of F, we have

$$\begin{aligned}
\mathbb{E}\left[\mathsf{F}(w_t, \mathsf{u}_t) - \mathsf{F}(w^*, \mathsf{u}^*)\right] &\leq \mathbb{E}\left[\langle \nabla \mathsf{F}(w_t, \mathsf{u}_t),\ (w_t, \mathsf{u}_t) - (w^*, \mathsf{u}^*)\rangle\right] && (16)\\
&= \mathbb{E}\left[\langle \nabla_w \mathsf{F}(w_t, \mathsf{u}_t), w_t - w^*\rangle\right] + \mathbb{E}\left[\langle \nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{F}(w_t, \mathsf{u}_t), \mathsf{u}_t^{\mathsf{Pub}} - \mathsf{u}^{\mathsf{Pub}*}\rangle\right] && (17)\\
&\quad + \mathbb{E}\left[\langle \nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{F}(w_t, \mathsf{u}_t), \mathsf{u}_t^{\mathsf{Priv}} - \mathsf{u}^{\mathsf{Priv}*}\rangle\right] && (18)
\end{aligned}$$

As in the standard analysis of gradient descent for convex objectives, we combine (18) with (13), (14), and (15), and use the fact that $\mathsf{F}(\bar{w}, \bar{\mathsf{u}}) - \mathsf{F}(w^*, \mathsf{u}^*) \leq \frac{1}{T}\sum_{t=1}^{T}(\mathsf{F}(w_t, \mathsf{u}_t) - \mathsf{F}(w^*, \mathsf{u}^*))$ (which follows from the convexity of F) to arrive at

$$\begin{aligned}
\mathbb{E}\left[\mathsf{F}(\bar{w}, \bar{\mathsf{u}}) - \mathsf{F}(w^*, \mathsf{u}^*)\right] \leq &\frac{\Lambda^2}{2\eta_w T} + \frac{\eta_w}{2}(G^2 + d\sigma_1^2) + \frac{U^{\mathsf{Priv}}}{2\eta_{\mathsf{u}^{\mathsf{Priv}}} T} + \frac{\eta_{\mathsf{u}^{\mathsf{Priv}}}}{2}\left(\frac{(1-\alpha)^4\bar{B}^2}{n^3} + n\sigma_2^2\right) \\
&+ \frac{U^{\mathsf{Pub}}}{2\eta_{\mathsf{u}^{\mathsf{Pub}}} T} + \frac{\eta_{\mathsf{u}^{\mathsf{Pub}}}}{2} \cdot \frac{\alpha^4(B + \bar{B})^2}{m^3}
\end{aligned}$$

Substituting with the choices of η_w, $\eta_{\mathsf{u}^{\mathsf{Pub}}}$, and $\eta_{\mathsf{u}^{\mathsf{Priv}}}$ in step 4 of Algorithm 1 yields the claimed upper bound on the expected optimization error. □

D. Proofs of Section 5

D.1. Proof of Lemma 5.1

Lemma D.1. *The objective function J admits the following properties:*
(i) J satisfies the same properties as those stated for F in Lemma 4.2.
(ii) Assume $m^{\frac{1}{3}} = O(n), n = O(m^3)$, and $\mu = O\left((m+n)^{\frac{2}{3}}\right)$. Then, J is $\bar{\beta}$-smooth over $\mathcal{V}$, where $\bar{\beta} = O(\beta)$.

Proof. First, the proof of item 1 is similar to that of Lemma 4.2 with minor, straightforward differences: first, note that that replacing the squared loss with any G-Lipschitz loss impacts neither the bounds on the norm of the gradient components nor the sensitivity of the gradients with respect to the private dataset; second, the two different terms in J are the $\left(1 - \sum_{i=1}^{m+n}\frac{1}{\mathsf{u}_i}\right)$ term and the μ-softmax term $\frac{1}{\mu}\log\left(\sum_{i=1}^{m+n} e^{\frac{\mu}{\mathsf{u}_i}}\right)$ do not affect the bounds on the gradient norms (a straightforward calculation of the gradients of these terms with respect to $\mathsf{u}^{\mathsf{Pub}}$ and $\mathsf{u}^{\mathsf{Priv}}$, together with the constraints on these variables, shows that the bounds on the norm of the gradient components still hold) and those two terms also do not have any effect on the sensitivity of the gradients with respect to the private dataset.

Next, we show the smoothness guarantee for J. First, note that J is twice differentiable. We can express its Hessian as

$$H(w, \mathsf{u}) = \begin{bmatrix} \nabla_w^2 \mathsf{J} & \mathsf{K} \\ \mathsf{K}^T & \nabla_{\mathsf{u}}^2 \mathsf{J} \end{bmatrix},$$

where, for any (w, u), $\nabla_w^2 \mathsf{J}(w, \mathsf{u}) \in \mathbb{R}^{d\times d}$ is given by $\nabla_w^2 \mathsf{J}(w, \mathsf{u}) = \left[\frac{\partial^2 \mathsf{J}}{\partial w_i \partial w_j}(w, \mathsf{u}) \colon (i,j) \in [d]^2\right]$, $\nabla_{\mathsf{u}}^2 \mathsf{J}(w, \mathsf{u}) \in \mathbb{R}^{(m+n)\times(m+n)}$ by $\nabla_{\mathsf{u}}^2 \mathsf{J}(w, \mathsf{u}) = \left[\frac{\partial^2 \mathsf{J}}{\partial \mathsf{u}_i \partial \mathsf{u}_j}(w, \mathsf{u}) \colon (i,j) \in [m+n]^2\right]$, and $\mathsf{K}(w, \mathsf{u}) \in \mathbb{R}^{d\times(m+n)}$ by $\mathsf{K}(w, \mathsf{u}) = \left[\frac{\partial^2 \mathsf{J}}{\partial w_i \partial \mathsf{u}_j}(w, \mathsf{u}) \colon i \in [d], j \in [m+n]\right]$. Note that the spectral norm of H can be upper bounded as follows:

$$\begin{aligned}
\|H(w, \mathsf{u})\|_2 &= \max_{\substack{V=(V_1,V_2)\\ \|V\|_2 \le 1}} \|H(w, \mathsf{u})V\|_2 \\
&= \max_{\substack{V=(V_1,V_2)\\ \|V\|_2 \le 1}} \left\| \begin{bmatrix} \nabla_w^2 \mathsf{J}(w, \mathsf{u})V_1 + \mathsf{K}(w, \mathsf{u})V_2 \\ \mathsf{K}^\top(w, \mathsf{u})V_1 + \nabla_u^2 \mathsf{J}(w, \mathsf{u})V_2 \end{bmatrix} \right\|_2 \\
&\le \|\nabla_w^2 \mathsf{J}(w, \mathsf{u})\|_2 + \|\mathsf{K}(w, \mathsf{u})\|_2 + \|\nabla_{\mathsf{u}}^2 \mathsf{J}(w, \mathsf{u})\|_2.
\end{aligned}$$

Thus, to prove that J is $O(\beta)$-smooth, it suffices for us to show that $\|\nabla_w^2 \mathsf{J}\|_2 + \|\mathsf{K}\|_2 + \|\nabla_{\mathsf{u}}^2 \mathsf{J}\|_2 = O(\beta)$.

First, observe that the following inequalities hold:

$$\left\|\nabla_w^2 \mathsf{J}\right\|_2 = \left\| \sum_{i=1}^{m+n} \nabla_w^2 \ell(w, x_i, y_i) \right\|_2 \le \sum_{i=1}^{m+n} \frac{\left\|\nabla_w^2 \ell(w, x_i, y_i)\right\|_2}{\mathsf{u}_i} \le \beta \sum_{i=1}^{m+n} \frac{1}{\mathsf{u}_i} \le \beta, \tag{19}$$

where the second inequality follows from the β-smoothness of the loss ℓ and the last inequality from the constraints $\mathsf{u}_i \ge \frac{m}{\alpha}$ for $i \in [m]$ and $\mathsf{u}_i \ge \frac{1-\alpha}{n}$ for $i \in [m+1, m+n]$.

Second, we bound $\left\|\nabla_{\mathsf{u}}^2 \mathsf{J}\right\|_2$. Observe that for all $i \in [m+n]$, we have

$$\frac{\partial^2 \mathsf{J}(w, \mathsf{u})}{\partial \mathsf{u}_i^2} = \frac{2\big(\ell(w, x_i, y_i) + \hat{d}_{\mathrm{DP}}\, 1_{i\le m} - \lambda_1\big)}{\mathsf{u}_i^3} + \lambda_2 \frac{3\frac{1}{\mathsf{u}_i^4}\sum_{j=1}^{m+n} \frac{1}{\mathsf{u}_j^2} - \frac{1}{\mathsf{u}_i^6}}{\left(\sum_{j=1}^{m+n} \frac{1}{\mathsf{u}_j^2}\right)^{\frac{3}{2}}} + \lambda_\infty \frac{\left(2 + \frac{\mu}{\mathsf{u}_i}\right)\frac{e^{\mu/\mathsf{u}_i}}{\mathsf{u}_i^3}\sum_{j=1}^{m+n} e^{\mu/\mathsf{u}_j} + \frac{\mu}{\mathsf{u}_i^4} e^{2\mu/\mathsf{u}_i}}{\left(\sum_{j=1}^{m+n} e^{\mu/\mathsf{u}_j}\right)^2}.$$

Thus, since the loss ℓ is uniformly bounded by B and $\mathsf{u}_i \ge \frac{m}{\alpha}$ $\forall i \in [m]$, we can bound $\left|\frac{\partial^2 \mathsf{J}(w,\mathsf{u})}{\partial \mathsf{u}_i^2}\right|$ for all $i \in [m]$ as follows:

$$\begin{aligned}
\left|\frac{\partial^2 \mathsf{J}(w, \mathsf{u})}{\partial \mathsf{u}_i^2}\right| &\le \frac{2\alpha^3|2B - \lambda_1|}{m^3} + 3\lambda_2 \frac{\frac{1}{\mathsf{u}_i^4}}{\sqrt{\sum_{j=1}^{m+n} \frac{1}{\mathsf{u}_j^2}}} + \lambda_\infty \frac{\frac{2}{\mathsf{u}_i^3} e^{\mu/\mathsf{u}_i} + \frac{\mu}{\mathsf{u}_i^4} e^{\mu/\mathsf{u}_i} \sum_{j=1}^{m+n} e^{\mu/\mathsf{u}_j}}{\left(\sum_{j=1}^{m+n} e^{\mu/\mathsf{u}_j}\right)^2} \\
&\le \frac{2\alpha^3|2B - \lambda_1|}{m^3} + \frac{3\lambda_2}{\mathsf{u}_i^3} + \lambda_\infty \left(\frac{2}{\mathsf{u}_i^3} + \frac{\mu}{\mathsf{u}_i^4}\right) \\
&\le \frac{2\alpha^3|2B - \lambda_1| + 3\lambda_2\alpha^3 + 2\lambda_\infty\alpha^3}{m^3} + \frac{\mu\lambda_\infty\alpha^4}{m^4}.
\end{aligned}$$

Similarly, we can show that for all $i \in [m+1, m+n]$,

$$\left|\frac{\partial^2 \mathsf{J}(w, \mathsf{u})}{\partial \mathsf{u}_i^2}\right| \le \frac{2(1-\alpha)^3|B - \lambda_1| + 3\lambda_2(1-\alpha)^3 + 2\lambda_\infty(1-\alpha)^3}{n^3} + \frac{\mu\lambda_\infty(1-\alpha)^4}{n^4}.$$

Moreover, for all $i, j \in [m+n]$ where $i \ne j$,

$$\begin{aligned}
\left|\frac{\partial^2 \mathsf{J}(w, \mathsf{u})}{\partial \mathsf{u}_i \partial \mathsf{u}_j}\right| &= \frac{\frac{\lambda_2}{\mathsf{u}_i^3 \mathsf{u}_j^3}}{\left(\sum_{t=1}^{m+n} \frac{1}{\mathsf{u}_t^2}\right)^{\frac{3}{2}}} + \frac{\frac{\lambda_\infty \mu}{\mathsf{u}_i^2 \mathsf{u}_j^2} e^{\mu/\mathsf{u}_i} e^{\mu/\mathsf{u}_j}}{\left(\sum_{t=1}^{m+n} e^{\mu/\mathsf{u}_t}\right)^2} \\
&\le \frac{\lambda_2}{\mathsf{u}_i^3} + \frac{\lambda_\infty \mu}{\mathsf{u}_i^2 \mathsf{u}_j^2} \\
&\le \begin{cases} \frac{\lambda_2\alpha^3}{m^3} + \frac{\lambda_\infty\mu\alpha^4}{m^4}, & i \in [m], j \in [m]\ (i \ne j) \\ \frac{\lambda_2\alpha^3}{m^3} + \frac{\lambda_\infty\mu\alpha^2(1-\alpha)^2}{m^2n^2}, & i \in [m], j \in [m+1, m+n] \\ \frac{\lambda_2(1-\alpha)^3}{n^3} + \frac{\lambda_\infty\mu\alpha^2(1-\alpha)^2}{m^2n^2}, & i \in [m+1, m+n], j \in [m] \\ \frac{\lambda_2(1-\alpha)^3}{n^3} + \frac{\lambda_\infty\mu(1-\alpha)^4}{n^4}, & i \in [m+1, m+n], j \in [m+1, m+n]\ (i \ne j) \end{cases}
\end{aligned}$$

Letting $\|\cdot\|_F$ denote the Frobenius norm, given all the above bounds, we can bound $\|\nabla^2_{\mathsf{u}}\mathsf{J}(w,\mathsf{u})\|_2$ as

$$\begin{aligned}\left\|\nabla^2_{\mathsf{u}}\mathsf{J}(w,\mathsf{u})\right\|_2 \le \left\|\nabla^2_{\mathsf{u}}\mathsf{J}(w,\mathsf{u})\right\|_F &= \left(\sum_{i=1}^{m+n}\sum_{j=1}^{m+n}\left|\frac{\partial^2\mathsf{J}(w,\mathsf{u})}{\partial\mathsf{u}_i\partial\mathsf{u}_j}\right|^2\right)^{\frac{1}{2}} \\ &\le \frac{\lambda_2\alpha^3}{m^2} + \frac{2\alpha^3\big(|2B-\lambda_1|+\lambda_2\sqrt{n}+\lambda_\infty\big)}{m^{\frac{5}{2}}} + \lambda_\infty\mu\alpha^4\left(\frac{1}{m^3}+\frac{1}{m^{\frac{7}{2}}}\right) \\ &+ \frac{\lambda_2(1-\alpha)^3}{n^2} + \frac{2(1-\alpha)^3\big(|B-\lambda_1|+\lambda_2\sqrt{m}+\lambda_\infty\big)}{n^{\frac{5}{2}}} + \lambda_\infty\mu(1-\alpha)^4\left(\frac{1}{n^3}+\frac{1}{n^{\frac{7}{2}}}\right) \\ &+ \frac{2\lambda_\infty\mu\alpha^2(1-\alpha)^2}{m^{\frac{3}{2}}n^{\frac{3}{2}}} \\ &\triangleq \beta'. \end{aligned} \tag{20}$$

Note that $\beta' = O\Big(\frac{\sqrt{n}}{m^{5/2}} + \frac{\sqrt{m}}{n^{5/2}} + \mu\big(\frac{1}{m^3}+\frac{1}{n^3}\big)\Big)$. Hence, when $m = O(n^3)$, $n = O(m^3)$, and $\mu = O\Big((m+n)^{\frac{2}{3}}\Big)$, we have $\beta' = O\big(\frac{1}{m}+\frac{1}{n}\big)$.

Finally, we bound $\|\mathsf{K}\|_2$. Observe that for all $i \in [d]$ and $j \in [m+n]$, we have

$$\left|\frac{\partial^2\mathsf{J}(w,\mathsf{u})}{\partial w_i\partial\mathsf{u}_j}\right| = \frac{\left|\frac{\partial\ell(w,x_j,y_j)}{\partial w_i}\right|}{\mathsf{u}_j^2}.$$

Hence, we have

$$\begin{aligned}\|\mathsf{K}\|_2 \le \|\mathsf{K}\|_F &= \left(\sum_{j=1}^{m+n}\frac{1}{\mathsf{u}_j^4}\sum_{i=1}^{d}\left|\frac{\partial\ell(w,x_j,y_j)}{\partial w_i}\right|^2\right)^{\frac{1}{2}} \\ &\le G\left(\sum_{j=1}^{m+n}\frac{1}{\mathsf{u}_j^4}\right)^{\frac{1}{2}} \\ &\le G\left(\frac{\alpha^2}{m^{3/2}} + \frac{(1-\alpha)^2}{n^{3/2}}\right). \end{aligned} \tag{21}$$

Putting inequalities (19), (20), and (21) together, we see that J is $\bar{\beta}$-smooth, where $\bar{\beta} = \beta + \beta' + G\Big(\frac{\alpha^2}{m^{3/2}} + \frac{(1-\alpha)^2}{n^{3/2}}\Big)$. As mentioned earlier, when the conditions on m, n, and μ in the lemma statement are satisfied, $\beta' = O\big(\frac{1}{m}+\frac{1}{n}\big)$. Thus, under these conditions, we have $\bar{\beta} = O(\beta)$. □

D.2. Formal description of Algorithm NCnvxAdap of Section 5

Next, we give the pseudocode for our private algorithm (Algorithm 2) for general adaptation scenarios described in Section 5.

D.3. Proof of Theorem 5.2

Theorem 5.2. *Algorithm* NCnvxAdap *is (ε,δ)-differentially private. Moreover, for the choice $T = O\Big(\varepsilon n/\sqrt{d\log(1/\delta)}\Big)$, the output of the algorithm satisfies the following bound on the norm of the gradient mapping: $\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(w_{t^*},\mathsf{u}_{t^*})\|_2^2 = O\Big(\sqrt{\bar{\beta}d\log(1/\delta)}/\varepsilon n\Big)$.*

Proof. First, we note that the privacy guarantee follows from exactly the same privacy argument for Algorithm 1 given in the proof of Theorem 4.3. This because the differences between our algorithm in Section 5 (Algorithm 2 in Appendix D.2) and Algorithm 1 do not impact the privacy analysis.

We now turn to the proof of convergence to a stationary point of J over $\mathcal{W}\times\mathcal{U}$ by showing that the expected norm of the gradient mapping of J at the output $(w_{t^*},\mathsf{u}_{t^*})$ is bounded as given in the theorem statement.

Algorithm 2 NCnvxAdap Private algorithm for general adaptation scenarios based on J

Require: $S^{\mathsf{Pub}} \in (\mathcal{X} \times \mathcal{Y})^m$; $S^{\mathsf{Priv}} \in (\mathcal{X} \times \mathcal{Y})^n$; privacy parameters (ε, δ); hyperparameters $\lambda_1, \lambda_2, \lambda_\infty$; number of iterations T.

1: Choose (w_0, u_0) in $\mathcal{W} \times \mathcal{U}$ arbitrarily.
2: Set $\sigma_1 := \frac{2s_1\sqrt{T\log(\frac{3}{\delta})}}{\varepsilon}$, where $s_1 := \frac{2(1-\alpha)G}{n}$.
3: Set $\sigma_2 := \frac{2s_2\sqrt{T\log(\frac{3}{\delta})}}{\varepsilon}$, where $s_2 := \frac{(1-\alpha)^2 B}{n^2}$.
4: Set step size $\eta := \frac{1}{\bar{\beta}}$, where $\bar{\beta}$ is the smoothness parameter given in Lemma 5.1.
5: **for** $t = 0$ to $T-1$ **do**
6: $\quad w_{t+1} := w_t - \eta(\nabla_w \mathsf{J}(w_t, \mathsf{u}_t) + \mathbf{z}_t)$, where $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \sigma_1^2 \mathbb{I}_d)$.
7: $\quad$ If $\|w_{t+1}\|_2 > \Lambda$ then $w_{t+1} \leftarrow \Lambda \frac{w_{t+1}}{\|w_{t+1}\|_2}$.
8: $\quad \mathsf{u}_{t+1}^{\mathsf{Pub}} := \mathsf{u}_t^{\mathsf{Pub}} - \eta \nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{J}(w_t, \mathsf{u}_t)$.
9: $\quad$ For every $i \in [m]$, set $\mathsf{u}_{i,t+1}^{\mathsf{Pub}} \leftarrow \max\left(\mathsf{u}_{i,t+1}^{\mathsf{Pub}}, \frac{m}{\alpha}\right)$.
10: $\quad \mathsf{u}_{t+1}^{\mathsf{Priv}} := \mathsf{u}_t^{\mathsf{Priv}} - \eta\left(\nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{J}(w_t, \mathsf{u}_t) + \mathbf{z}_t'\right)$, where $\mathbf{z}_t' \sim \mathcal{N}(\mathbf{0}, \sigma_2^2 \mathbb{I}_n)$.
11: $\quad$ For every $i \in [n]$, set $\mathsf{u}_{i,t+1}^{\mathsf{Priv}} \leftarrow \max\left(\mathsf{u}_{i,t+1}^{\mathsf{Priv}}, \frac{n}{1-\alpha}\right)$.
12: **end for**
13: **return** $(w_{t^*}, \mathsf{u}_{t^*})$, where t^* is uniformly sampled from $[T]$.

To simplify notation, we let $\mathsf{v}_t \triangleq (w_t, \mathsf{u}_t)$ $\forall t \in [T]$, let $\mathcal{V} \triangleq \mathcal{W} \times \mathcal{U}$, and let $\mathbf{g}_t \triangleq (\mathbf{z}_t, \mathbf{0}^m, \mathbf{z}_t')$ be the combined noise vector added to $\nabla \mathsf{J} = (\nabla_w \mathsf{J}(\mathsf{v}_t), \nabla_{\mathsf{u}^{\mathsf{Pub}}} \mathsf{J}(\mathsf{v}_t), \nabla_{\mathsf{u}^{\mathsf{Priv}}} \mathsf{J}(\mathsf{v}_t))$ in the t-th iteration $\forall t \in [T]$. Here, $\mathbf{0}^m$ denote the m-dimensional all-zero vector. Recall that $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}^d, \sigma_1^2 \mathbb{I}_d)$ and $\mathbf{z}_t' \sim \mathcal{N}(\mathbf{0}^n, \sigma_2^2 \mathbb{I}_n)$ as defined in steps 4 and 10 in Algorithm 2. We let $\bar{\sigma}^2 = d\sigma_1^2 + n\sigma_2^2$. Also, we let $\widetilde{\nabla}_t \triangleq \nabla \mathsf{J}(\mathsf{v}_t) + \mathbf{g}_t$, $\forall t \in [T]$. For any $v \in \mathbb{R}^{d+m+n}$, we let $\mathrm{Proj}_{\mathcal{V}}(v)$ denote the Euclidean projection of v onto $\mathcal{V}$.

By $\bar{\beta}$-smoothness of J, we have

$$\begin{aligned}
\mathsf{J}(\mathsf{v}_{t+1}) &\le \mathsf{J}(\mathsf{v}_t) + \langle \nabla \mathsf{J}(\mathsf{v}_t), \mathsf{v}_{t+1} - \mathsf{v}_t \rangle + \frac{\bar{\beta}}{2}\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2 \\
&= \mathsf{J}(\mathsf{v}_t) + \langle \widetilde{\nabla}_t, \mathsf{v}_{t+1} - \mathsf{v}_t \rangle - \langle \mathbf{g}_t, \mathsf{v}_{t+1} - \mathsf{v}_t \rangle + \frac{\bar{\beta}}{2}\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2
\end{aligned} \tag{22}$$

Note that $\mathsf{v}_{t+1} = \mathrm{Proj}_{\mathcal{V}}\left(\mathsf{v}_t - \eta \widetilde{\nabla}_t\right)$. By a known property of Euclidean projection (e.g., see (Beck, 2017)[Theorem 9.8]), we have

$$\langle \mathsf{v}_t - \eta \widetilde{\nabla}_t - \mathsf{v}_{t+1}, \mathsf{v}_t - \mathsf{v}_{t+1} \rangle \le 0,$$

which implies

$$\langle \widetilde{\nabla}_t, \mathsf{v}_t - \mathsf{v}_{t+1} \rangle \le -\frac{1}{\eta}\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2.$$

Hence, inequality (22) implies

$$\begin{aligned}
\mathsf{J}(\mathsf{v}_{t+1}) &\le \mathsf{J}(\mathsf{v}_t) - \frac{1}{\eta}\left(1 - \frac{\bar{\beta}\eta}{2}\right)\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2 - \langle \mathbf{g}_t, \mathsf{v}_{t+1} - \mathsf{v}_t \rangle \\
&\le \mathsf{J}(\mathsf{v}_t) - \frac{1}{\eta}\left(1 - \frac{\bar{\beta}\eta}{2}\right)\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2 + \|\mathbf{g}_t\|_2 \|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2.
\end{aligned}$$

By setting $\eta = \frac{1}{\bar{\beta}}$, taking the expectation of both sides of the inequality above, and use the fact that $\mathbb{E}\left[\|\mathbf{g}_t\|_2 \|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2\right] \le \sqrt{\mathbb{E}\left[\|\mathbf{g}_t\|_2^2\right] \mathbb{E}\left[\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2\right]}$, we get

$$\mathbb{E}\left[\mathsf{J}(\mathsf{v}_{t+1})\right] \le \mathbb{E}\left[\mathsf{J}(\mathsf{v}_t)\right] - \frac{\bar{\beta}}{2}\left(\sqrt{\mathbb{E}\left[\|\mathsf{v}_{t+1} - \mathsf{v}_t\|_2^2\right]} - \frac{\bar{\sigma}}{\bar{\beta}}\right)^2 + \frac{\bar{\sigma}^2}{2\bar{\beta}},$$

which implies

$$\left(\sqrt{\mathbb{E}\left[\bar{\beta}^2\|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2^2\right]}-\bar{\sigma}\right)^2 \le 2\bar{\beta}\mathbb{E}\left[\mathsf{J}(\mathsf{v}_{t+1})-\mathsf{J}(\mathsf{v}_t)\right]+\bar{\sigma}^2.$$

Since $\mathbb{E}\left[\bar{\beta}^2\|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2^2\right] \le 2\left(\sqrt{\mathbb{E}\left[\bar{\beta}^2\|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2^2\right]}-\bar{\sigma}\right)^2+2\bar{\sigma}^2$, we get

$$\mathbb{E}\left[\bar{\beta}^2\|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2^2\right] \le 4\bar{\beta}\mathbb{E}\left[\mathsf{J}(\mathsf{v}_{t+1})-\mathsf{J}(\mathsf{v}_t)\right]+4\bar{\sigma}^2. \tag{23}$$

For any $t \in [T]$, let $\mathsf{v}_t^\dagger \triangleq \mathrm{Proj}_{\mathcal{V}}(\mathsf{v}_t-\eta\nabla\mathsf{J}(\mathsf{v}_t))$. Observe that

$$\begin{aligned}
\|\mathsf{v}_t-\mathsf{v}_t^\dagger\|_2 &\le \|\mathsf{v}_t-\mathsf{v}_{t+1}\|_2+\|\mathsf{v}_{t+1}-\mathsf{v}_t^\dagger\|_2\\
&= \|\mathsf{v}_t-\mathsf{v}_{t+1}\|_2+\|\mathrm{Proj}_{\mathcal{V}}\big(\mathsf{v}_t-\eta\widetilde{\nabla}_t\big)-\mathrm{Proj}_{\mathcal{V}}(\mathsf{v}_t-\eta\nabla\mathsf{J}(\mathsf{v}_t))\|_2\\
&\le \|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2+\eta\|\mathbf{g}_t\|_2\\
&= \|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2+\frac{\|\mathbf{g}_t\|_2}{\bar{\beta}},
\end{aligned}$$

where the first bound follows from the triangle inequality and the third bound follows from the non-expansiveness of the Euclidean projection. Thus, we have

$$\mathbb{E}\left[\|\mathsf{v}_t-\mathsf{v}_t^\dagger\|_2^2\right] \le 2\mathbb{E}\left[\|\mathsf{v}_{t+1}-\mathsf{v}_t\|_2^2\right]+2\frac{\bar{\sigma}^2}{\bar{\beta}^2}.$$

Combining this with (23) yields

$$\mathbb{E}\left[\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(\mathsf{v}_t)\|_2^2\right]=\mathbb{E}\left[\bar{\beta}^2\|\mathsf{v}_t-\mathsf{v}_t^\dagger\|_2^2\right] \le 8\bar{\beta}\mathbb{E}\left[\mathsf{J}(\mathsf{v}_{t+1})-\mathsf{J}(\mathsf{v}_t)\right]+10\bar{\sigma}^2.$$

Now, taking expectation with respect to the randomness in the uniformly drawn index t^* of the output, we get

$$\begin{aligned}
\mathbb{E}\left[\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(\mathsf{v}_{t^*})\|_2^2\right] &= \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}\left[\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(\mathsf{v}_t)\|_2^2\right]\\
&\le \frac{8\bar{\beta}}{T}\mathbb{E}\left[\mathsf{J}(\mathsf{v}_0)-\mathsf{J}(\mathsf{v}_T)\right]+10\bar{\sigma}^2\\
&\le 8\left(\frac{\bar{\beta}M}{T}+\frac{40(1-\alpha)^2G^2dT\log(2/\delta)}{\varepsilon^2n^2}+\frac{(1-\alpha)^4B^2T\log(2/\delta)}{\varepsilon^2n^3}\right),
\end{aligned}$$

where in the second inequality, we use the fact that J is uniformly bounded over $\mathcal{V}$ by $M \triangleq 2B+\lambda_1+\lambda_2\left(\frac{\alpha}{\sqrt{m}}+\frac{1-\alpha}{\sqrt{n}}\right)+\lambda_\infty\max\left(\frac{\alpha}{m},\frac{1-\alpha}{n}\right)$. By setting

$$T=\frac{\sqrt{\bar{\beta}M}\varepsilon n^{3/2}}{(1-\alpha)\sqrt{\log(\frac{2}{\delta})}\sqrt{40G^2dn+(1-\alpha)^2B^2}}=O\left(\frac{n\varepsilon}{\sqrt{d\log(\frac{1}{\delta})}}\right),$$

we finally obtain

$$\mathbb{E}\left[\|\mathcal{G}_{\mathsf{J},\bar{\beta}}(\mathsf{v}_{t^*})\|_2^2\right] \le 16(1-\alpha)\sqrt{\bar{\beta}M}\sqrt{\log\left(\frac{2}{\delta}\right)}\left(\frac{2\sqrt{10}G\sqrt{d}}{\varepsilon n}+\frac{(1-\alpha)B}{\varepsilon n^{\frac{3}{2}}}\right)=O\left(\frac{\sqrt{\bar{\beta}d\log\left(\frac{1}{\delta}\right)}}{\varepsilon n}\right),$$

which completes the proof. □

Table 3. MSE of non-private convex algorithm against unsupervised baselines. We report relative errors normalized so that training on target only has an MSE of 1.0. The best results are indicated in boldface.

Dataset	KMM	DM	$\mathsf{CnvxAdap}_\infty$
`Wind`	1.009 ± 0.035	1.009 ± 0.045	$\mathbf{0.985 \pm 0.019}$
`Airline`	2.716 ± 0.202	1.547 ± 0.068	$\mathbf{0.992 \pm 0.012}$
`Gas`	0.441 ± 0.034	0.381 ± 0.028	$\mathbf{0.342 \pm 0.023}$
`News`	1.162 ± 0.044	1.006 ± 0.009	$\mathbf{0.995 \pm 0.019}$
`Slice`	1.282 ± 0.076	1.218 ± 0.130	$\mathbf{0.992 \pm 0.050}$

Dataset	Target training sample size	Target validation sample size	Target test sample size	Source training sample size	Input dimension
`Wind`	158	200	200	6016	11
`Airline`	200	300	300	16000	11
`Gas`	613	1000	2000	7297	133
`News`	737	1000	1000	3000	50
`Slice`	1077	154	308	7803	384

Table 4. Sample sizes of the datasets `Wind`, `Airline`, `Gas`, `News` and `Slice`.

E. Additional experimental results

E.1. Convex setting

Table 4 gives the sample sizes and input dimensions of the datasets `Wind`, `Airline`, `Gas`, `News` and `Slice`. For the `Wind` dataset (Haslett & Raftery, 1989), the source and target data are collected in different months of the year with the labels being the speed of the wind. The `Airline` dataset stems from (Ikonomovska, 2009). The source and target data come from a subset of the data for the Chicago O'Haire International Airport (ORD) in 2008 and are divided based on different hours of the day. The goal is to predict the amount of time the flight is delayed. The `Gas` dataset (Rodriguez-Lujan et al., 2014; Vergara et al., 2012; Dua & Graff, 2017) uses features from various sensor measurements to predict the concentration level. For the `News` dataset (Fernandes, 2015; Dua & Graff, 2017), the source contains articles from Monday to Saturday while the target contains articles from Sunday. The task is to predict the popularity of the articles. The `Slice` dataset (Graf et al., 2011) uses features retrieved from CT images to predict the relative location of CT slices on the axial axis of the human body. The source and target data are divided based on individual patients.

We compare our non-private convex algorithm $\mathsf{CnvxAdap}_\infty$ with unsupervised domain adaptation baselines, the Kernel Mean Matching (KMM) algorithm (Huang et al., 2006) and the DM algorithm (Cortes & Mohri, 2014). Table 3 shows that our convex algorithm consistently outperforms the baselines. For reference, we also compared our algorithm with the noisy minibatch SGD from (Bassily et al., 2019) (top dash-dotted plots in the figure), see Figure 2. We verify that our algorithm outperforms that noisy minibatch SGD algorithm, which only benefits from the target labeled data.

To further illustrate the effectiveness of our algorithms, we also report a series of additional empirical results comparing our private adaptation algorithm $\mathsf{CnvxAdap}$ to the non-private baseline, the DM algorithm (Cortes & Mohri, 2014), on the multi-domain sentiment analysis dataset (Blitzer et al., 2007) formed as a regression task for each category as in prior work (Awasthi et al., 2024). We consider four categories: `BOOKS`, `DVD`, `ELECTRONICS`, and `KITCHEN`. We report MeanSquaredErrors, MSE, for 12 pairwise experiments (TaskA, TaskB) in Table 5. As a source, we use a combination of 500 examples from TaskA and 200 examples from TaskB. For the target data we use 300 examples from TaskB. We use 50 examples from TaskB for validation and 1000 examples for testing. The results are averaged over 10 independent source/target splits, which show that our private adaptation algorithm $\mathsf{CnvxAdap}$ consistently outperforms DM (non-private algorithm), even for $\varepsilon = 10$ for this relatively small target sample size.

For a more fair comparison with private-DM (Bassily et al., 2022), here, we also provide results for our private algorithm $\mathsf{CnvxAdap}$ with $\alpha = 1$. In future work we seek to estimate α in a principled way based on the discrepancy. Figure 3 shows that even for high values of n, the private-DM does not outperform our algorithm.

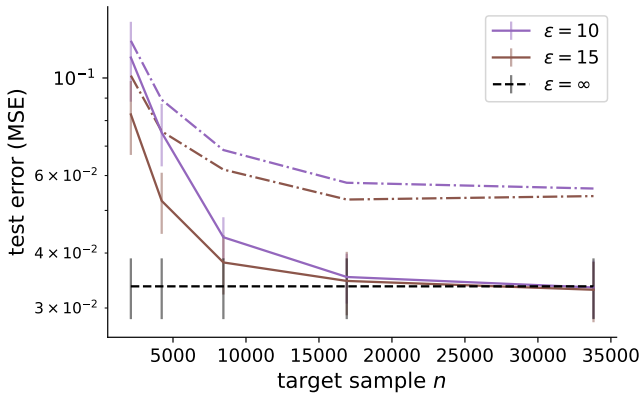

Figure 2. MSE on the `Gas` dataset over ten runs against the number of target samples with various values of ϵ. Comparison of our algorithm (solid lines) with the noisy minibatch SGD from (Bassily et al., 2019) (dash-dotted lines).

Table 5. MSE of CnvxAdap for $\epsilon = 4$ and $\epsilon = 10$ against the non-private DM algorithm on the sentiment analysis dataset.

Source	Target	CnvxAdap (ε = 4)	CnvxAdap (ε = 10)	DM
	BOOKS	1.649 ± 1.909	0.713 ± 0.625	0.778 ± 0.459
KITCHEN	DVD	0.832 ± 0.640	0.521 ± 0.354	1.396 ± 1.145
	ELEC	0.790 ± 0.153	0.555 ± 0.139	1.740 ± 1.483
	DVD	0.929 ± 0.365	0.637 ± 0.297	0.761 ± 0.562
BOOKS	ELEC	0.708 ± 0.092	0.485 ± 0.147	0.711 ± 0.320
	KITCHEN	0.796 ± 0.126	0.632 ± 0.181	0.956 ± 0.442
	ELEC	0.671 ± 0.120	0.429 ± 0.189	0.677 ± 0.231
DVD	KITCHEN	0.665 ± 0.197	0.453 ± 0.189	1.185 ± 0.753
	BOOKS	0.727 ± 0.288	0.430 ± 0.143	0.676 ± 0.812
	KITCHEN	0.671 ± 0.161	0.439 ± 0.141	1.389 ± 0.755
ELEC	BOOKS	0.743 ± 0.365	0.351 ± 0.204	0.884 ± 0.639
	DVD	0.755 ± 0.585	0.589 ± 0.495	0.843 ± 0.403

E.2. Non-convex setting

The detailed information about the sample sizes and input dimensions of the datasets `Adult`, `German`, `Accent`, `CIFAR-100`, `CIFAR-10`, `SVHN` and `ImageNet` is given in Table 7. For the `Adult` dataset, also known as the Census Income dataset, the source and target data are divided by the gender attribute to predict whether the income exceeds \$50K. The source and target data of the South German Credit dataset, `German`, are divided based on whether the debtor has lived in the present residence for at least three years. The goal is to predict the status of the debtor's checking account with the bank. The Speaker Accent Recognition dataset, `Accent`, uses features from the soundtrack of words read by speakers from different countries to predict the accent. The source contains examples whose language attribute is US or UK while the target contains the remaining examples.

We use the CLIP (Radford et al., 2021) model to extract features from the `ImageNet` (Deng et al., 2009) dataset and extract features from the `CIFAR-100`, `CIFAR-10` (Krizhevsky, 2009) and `SVHN` (Netzer et al., 2011) datasets by using the outputs of the second-to-last layer of ResNet (He et al., 2016). We transform those datasets into binary classification by assigning half of the labels as $+1$ and the other half as -1 and then convert them into domain adaptation tasks where the source and target data consist of distinct mixtures of uniform sampling and Gaussian sampling using the mean and covariance of the data. For the `CIFAR-100` and `ImageNet` datasets, 95% of the source data and 5% of the target data come from Gaussian sampling; for the `CIFAR-10` dataset, 90% of the source data and 10% of the target data come from Gaussian sampling; for the `SVHN` dataset, 80% of the source data and 20% of the target data come from Gaussian sampling.

We compare our non-private non-convex algorithm $\mathsf{NCnvxAdap}_\infty$ with the KMM algorithm (Huang et al., 2006). Table 6 shows that our non-convex algorithm consistently outperforms the baselines.

Hyperparameter tuning. We do hyper-parameter selection based on a validation set, while the reported results are

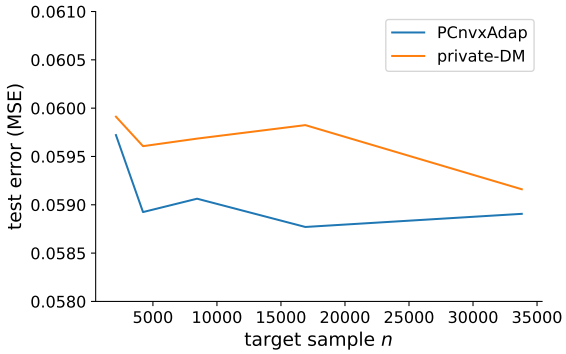

Figure 3. Mean values of MSE on the `Gas` dataset over ten runs against the number of target samples with $\epsilon = 15$. Comparison of our algorithm ($\alpha = 1$) with the private-DM from (Bassily et al., 2022).

Table 6. Accuracy of non-private non-convex algorithm against unsupervised baselines. Best results are boldfaced.

Dataset	Train source	Train target	KMM	NCnvxAdap$_\infty$
`Adult`	88.44 ± 0.61	91.09 ± 0.54	74.62 ± 0.64	$\mathbf{91.38 \pm 0.49}$
`German`	74.16 ± 1.64	76.40 ± 1.17	74.43 ± 1.65	$\mathbf{77.96 \pm 1.31}$
`Accent`	51.60 ± 1.09	92.27 ± 1.10	52.18 ± 2.27	$\mathbf{93.95 \pm 0.78}$
`CIFAR-100`	58.50 ± 0.30	59.51 ± 0.38	58.27 ± 0.28	$\mathbf{62.19 \pm 0.32}$
`CIFAR-10`	83.18 ± 0.47	85.18 ± 0.24	83.57 ± 0.45	$\mathbf{86.44 \pm 0.29}$
`SVHN`	84.16 ± 1.21	86.28 ± 0.44	84.13 ± 1.21	$\mathbf{87.34 \pm 0.46}$
`ImageNet`	71.20 ± 1.75	85.70 ± 0.29	71.25 ± 1.77	$\mathbf{86.84 \pm 0.38}$

evaluated on the test set. The sample sizes of the datasets are reported in Table 4 and Table 7. We select α over $\{0.1, 0.3, 0.5, 0.7, 0.9\}$. For Algorithm 1, we select κ_1 over $\{1, 5, 10, 100, 200\}$, κ_2 over $\{5, 10, 50, 100, 1000\}$, κ_∞ over $\{0.1, 1, 5, 10, 100\}$ and the number of iterations T is set as 15,000. For Algorithm 2, we select λ_1 over $\{1, 10, 100, 200, 250\}$, λ_2 over $\{0.01, 0.1, 1, 10, 100\}$, λ_∞ over $\{0.01, 0.1, 1, 5, 10\}$ and the number of iterations T is set as 15,000. While our algorithm involves hyperparameters, it is important to note that this holds for virtually all standard learning algorithms, even in the absence of adaptation; e.g., neural networks require fine-tuning of multiple parameters through validation datasets. In particular, our methodology does not rely more on hyperparameter tuning than the baselines. Empirical hyperparameter tuning can incur a privacy cost indeed, which requires careful attention. One approach to mitigate this is using privacy-preserving hyperparameter tuning via local sensitivity analysis to estimate the privacy cost of each tuning query.

Sampling with replacement. We use sampling with replacement to increase the number of samples in the datasets we experiment on and enable reporting results for larger values of target sample size n. In terms of the privacy guarantee, we would like to note that the sampling with replacement we perform does not mean that each individual in the resulting dataset contributes multiple data points. Each of the repeated data points is viewed as belonging to a different individual (that is, we assume the total number of individuals in the dataset also increases with sampling so that the size of the sampled dataset equals the total number of individuals).

Dataset	Target training sample size	Target validation sample size	Target test sample size	Source training sample size	Input dimension
Adult	6847	978	1957	20379	14
German	2452	350	702	562	20
Accent	833	119	238	210	12
CIFAR-100	250	8250	16500	2500	64
CIFAR-10	125	4125	8251	1249	64
SVHN	122	4029	8058	1830	64
ImageNet	128	4227	8456	1280	512

Table 7. Sample sizes of the datasets Adult, German, Accent, CIFAR-100, CIFAR-10, SVHN and ImageNet.