Differentially Private Model Personalization

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study personalization of supervised learning with user-level differential privacy. Consider a setting with many users, each of whom has a training data set drawn from their own distribution P_i. Assuming some shared structure among the problems P_i, can users collectively learn the shared structure—and solve their tasks better than they could individually—while preserving the privacy of their data? We formulate this question using joint, *user-level* differential privacy—that is, we control what is leaked about each user's entire data set.

We provide algorithms that exploit popular non-private approaches in this domain like the Almost-No-Inner-Loop (ANIL) method, and give strong user-level privacy guarantees for our general approach. When the problems P_i are linear regression problems with each user's regression vector lying in a common, unknown low-dimensional subspace, we show that our efficient algorithms satisfy nearly optimal estimation error guarantees. We also establish a general, information-theoretic upper bound via an exponential mechanism-based algorithm.

1 Introduction

Modern machine learning techniques are amazingly successful but come with a range of risks to the privacy of the personal data on which they are trained. Complex models often encode exact personal information in surprising ways—allowing, in extreme cases, the exact recovery of training data from black box use of the model [6, 7]. The emerging architecture of modern learning systems, in which models are trained collaboratively by networks of mobile devices using extremely rich, personal information exacerbates these risks.

The paradigm of *model personalization*, a special case of multitask learning, has emerged as one way to address both privacy and scalability issues. The idea is to let users train models on their own data—for example, to recognize friends' and family members' faces in photos, or to suggest text completions that match the user's style—based on information that is common to the many other similar learning problems being solved by other users in the system. Even a fairly limited amount of shared information—a useful feature representation or starting set of parameters for optimization, for example—can dramatically reduce the amount of data each user requires. But that shared information can nevertheless be highly disclosive.

In this paper, we formulate a model for reasoning rigorously about the loss to privacy incurred by sharing information for model personalization. In our model, there are n users, each holding a dataset of m labeled examples. We assume user j's data set D_j is drawn i.i.d. from a distribution P_j; the user's goal is to learn a prediction rule that generalizes well to unseen examples from P_j. Ideally, the user should succeed much better than they could have on their own. We give new algorithms for this setting, analyze their accuracy on specific data distributions, and test our results empirically.

We ask that our algorithms satisfy *user-level, joint differential privacy* (DP) [27] (called *task-level* privacy, in the context of multi-task learning [31]). In this setting, each user provides their data set D_j as input to the algorithm and receives output $A_j = A_j(D_1, ..., D_n)$. We require that for every

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

choice of the other data sets $D_{-j} = (D_1, ..., D_{j-1}, D_{j+1}, ..., D_n)$ and for every two data sets D_j and D'_j, the collective view of the other users A_{-j} be distributed essentially identically regardless of whether user j inputs D_j or D'_j. The standard model of differential privacy doesn't directly fit our setting, since the model ultimately trained by user j will definitely reveal information about user j's data set. That said, the algorithms we design can ultimately be viewed as an appropriate composition of modules that satisfy the usual notion of DP (an approach known as the *billboard* model). For simplicity, we describe our algorithms in a centralized model in which the data are stored in a single location, and the algorithm $\mathcal{A}$ is run as a single operation. In most cases, we expect $\mathcal{A}$ to be run as a distributed protocol, using either general tools such as multiparty computation or lightweight, specialized ones such as differentially private aggregation to simulate the shared platform.

Intuitively, strong privacy requirement at user level, while still demanding that users share some common information is significantly challenging. For one, as each user individually has a small amount of data, it has to share information about it's model/data to learn a meaningful representation. Furthermore, in practical personalization settings, there is feedback loop between the common or *pooled* knowledge of all users and the personalized models for each user. That is, starting with reasonable personalized models for each user, leads to a better pooled information, while good pooled information then helps each user learn better personal model. Now, requirement of strong privacy guarantees forces the pooled information quality to degrade up to some extent, which can then lead to poorer personalized model and form a negative feedback loop.

1.1 Contributions

We consider two types of algorithms for DP model personalization: inefficient algorithms (based on the exponential mechanism [34]) that establish information-theoretic upper bounds on achievable error, and efficient ones based on popular iterative approaches to non-private personalization [39, 25, 50, 51]. These latter approaches are popular for their convergence speed and low communication overhead. As is often the case, those same features make them attractive starting points for DP algorithms.

Problem Setting: Consider a set of n users, and suppose each user $j \in [n]$ holds a data set of m records $D_j = \{(\mathbf{x}_{ij}, y_{ij})\}_{i\in[m]}$ where $\mathbf{x}_{ij} \in \mathbb{R}^d$, $y_{ij} \in \mathbb{R}$. The goal is to learn a personalized model $f_j(\cdot) = f(\cdot; \theta_j) : \mathbb{R}^d \to \mathbb{R}$ for each user j, where θ_j is a vector of parameters describing the model.

We aim to learn a shared, low-dimensional representation for the features that allows users to train good predictors individually. For concreteness, we consider a linear embedding specified by a $d \times k$ matrix $\boldsymbol{U}$, where $k \ll d$. We may think of $\boldsymbol{U}$ either as providing a k-dimensional representation of the feature $\mathbf{x}_{ij}$ (as $\boldsymbol{U}^\top \mathbf{x}_{ij}$) or, alternatively, as a compact way to specify a d-dimensional regression vector $\theta_j = \boldsymbol{U}\boldsymbol{v}_j$ where $\boldsymbol{v}_j$ is vector of length k. In both cases, user j's final predictor has the form

$$f_j(\mathbf{x}_{ij}) = f'(\langle \mathbf{x}_{ij}, \boldsymbol{U}\boldsymbol{v}_j\rangle) = f'(\langle \boldsymbol{U}^\top \mathbf{x}_{ij}, \boldsymbol{v}_j\rangle)$$

One may view this as a model as a two-layer neural network, where the first layer is shared across all users and the second layer is trained individually. A useful setting to have in mind is one where $k \ll m \ll d$—so users do not have enough data to find a good solution on their own, but they do have enough data to find the best vector $\boldsymbol{v}_j$ once an embedding $\boldsymbol{U}$ has been specified. Without loss of generality, we assume $\boldsymbol{U} \in \mathbb{R}^{d\times k}$ to be an orthonormal basis and refer to it as *embedding matrix*. For brevity, we will define the matrix $\boldsymbol{V} = [\boldsymbol{v}_1 | \cdots | \boldsymbol{v}_n] \in \mathcal{C} \subseteq \mathbb{R}^{k\times n}$ with $\boldsymbol{v}_j$s as columns.

Measure of Accuracy: Let $\mathcal{L}_{\texttt{Pop}}(\boldsymbol{U}; \boldsymbol{V}) = \mathbb{E}_{(i,j)\sim_u [m]\times[n], (\mathbf{x}_{ij}, y_{ij})\sim P_j}\left[\ell\left(\langle \boldsymbol{U}^\top \mathbf{x}_{ij}, \boldsymbol{v}_j\rangle; y_{ij}\right)\right]$, where the loss function takes the form $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$. We will focus on excess population risk defined in (1). The privately learned models are denoted by $\left(\boldsymbol{U}^{\texttt{priv}}, \boldsymbol{V}^{\texttt{priv}}\right)$. The error measures are defined with respect to any fixed choice of parameters $(\boldsymbol{U}^*, \boldsymbol{V}^*)$.

$$\mathsf{Risk}_{\texttt{Pop}}(\left(\boldsymbol{U}^{\texttt{priv}}, \boldsymbol{V}^{\texttt{priv}}\right); (\boldsymbol{U}^*, \boldsymbol{V}^*)) = \mathcal{L}_{\texttt{Pop}}(\boldsymbol{U}^{\texttt{priv}}, \boldsymbol{V}^{\texttt{priv}}) - \mathcal{L}_{\texttt{Pop}}(\boldsymbol{U}^*, \boldsymbol{V}^*). \quad (1)$$

Alternating Minimization Framework: We develop an efficient framework based on *alternating minimization* [45, 28, 22]: starting from an initial embedding map $\boldsymbol{U}_0$, the algorithm proceeds in rounds that alternate between users individually selecting the model $\boldsymbol{v}_j^{(t)}$ that minimizes the error of the predictor $f'(\langle \cdot, \boldsymbol{U}^{(t)}\boldsymbol{v}_j^{(t)}\rangle)$, and then running a DP algorithm, for which user j provides inputs

$D_j, \boldsymbol{v}_j^{(t)}$, to privately select a new embedding $\boldsymbol{U}^{(t+1)}$ that minimizes the error of the predictor $f'(\langle \cdot, \boldsymbol{U}^{(t+1)} \boldsymbol{v}_j^{(t)} \rangle)$. In both steps, the optimization to be performed is convex when the loss being optimized is convex. This helps us handle the inherent non-convexity in the problem formulation.

Instantiation and Analysis for Linear Regression with Gaussian Data: For the specific case of linear regression with the squared error loss, we show that our framework can be fully instantiated with an efficient algorithm which converges quickly to an optimal solution. For simplicity, we consider the case where the feature vectors and field noise are normally distributed and independent of each user's "true" model θ_j^*, and furthermore that the θ_j^* vectors admit a common low-dimensional representation $\boldsymbol{U}^* \in \mathbb{R}^{d \times k}$, so that $\theta_j^* = \boldsymbol{U}^* \boldsymbol{v}_j^*$. We show that careful initialization of $\boldsymbol{U}_0$ followed by alternating minimization converges to a near-optimal embedding as long as $m = \omega(k^2)$ and $n = \omega\left(\frac{k^{2.5} d^{1.5}}{\varepsilon}\right)$. Notice that non-privately, one would require $n = \omega(dk)$ users to get any reasonable test error. For standard *private* linear regression in dk dimensions, current state-of-the-art results (Theorem 3.2, [3]) have a sample complexity similar to what we achieve.

Theorem 1.1 (Informal version of Theorem 4.2)**.** *Suppose the output for point* $\mathbf{x}_{ij} \sim \mathcal{N}(0,1)^d$ *of user-*j *is given by:* $y_{ij} \sim \langle (\boldsymbol{U}^*)^\top \mathbf{x}_{ij}, \boldsymbol{v}_j^* \rangle + \mathcal{N}(0, \sigma_F^2)$ *where* $\boldsymbol{U}^*$ *parameterizes the shared representation. For simplicity, suppose* $\boldsymbol{v}_j^* \sim \mathcal{N}(0,1)^k$*. Then, assuming the number of users* $n \geq (kd)^{1.5}/\varepsilon$*, Algorithm 1 learns an embedding matrix* $\boldsymbol{U}^{priv}$ *s.t. the average test error of a linear regressor learned over points embedded by* $\boldsymbol{U}^{priv}$ *is at most* $\widetilde{O}\left(\frac{(\sigma_F^2 + dk^2)(dk)^2 \cdot k}{\varepsilon^2 n^2} + \sigma_F^2 \left(\frac{dk^2}{mn} + \frac{k}{m}\right)\right)$.

Our instantiation of the framework in this case has two major components: The initial embedding $\boldsymbol{U}_0$ is derived from users' data by a single noisy averaging step which roughly approximates the $d \times d$ projector onto the k-dimensional column space of $\boldsymbol{U}^*$. The idea is that given two data points $(\mathbf{x}_{ij}, y_{ij})$ and $(\mathbf{x}_{(i+1)j}, y_{(i+1)j})$, the expected value of the rank-one matrix $y_{ij} y_{(i+1)j} \mathbf{x}_{ij} \mathbf{x}_{(i+1)j}^\top$ is (when rescaled) a projector onto the space spanned by the regression vector θ_j. Adding these rank-one matrices across many data points and users produces a matrix with high overlap with the desired projector $\boldsymbol{U}^* (\boldsymbol{U}^*)^\top$. This is similar to the approach taken by [12] to design a non-private algorithm for a related, less general setting.

The DP minimization step, which fixes the $\boldsymbol{v}_j$'s and seeks a near-minimal $\boldsymbol{U}$, can be performed using any DP algorithm for convex minimization [8, 4]. In this particular case, one can view this step as solving a linear regression problem in which $\boldsymbol{U}$ represents a list of dk real parameters: once $\mathbf{x}$ and $\boldsymbol{v}$ are fixed, $\langle \boldsymbol{U}^\top \mathbf{x}, \boldsymbol{v} \rangle = \mathbf{x}^\top \boldsymbol{U} \boldsymbol{v}$ is a linear function of $\boldsymbol{U}$.

For the analysis to be tractable, we restrict our attention to linear regression with independent, normally-distributed features. However, the framework we provide is more general, and can be applied to a wider class of models. Developing mathematical tools to analyse the behavior of noisy alternating minimization algorithms in more general settings remains an important open question.

Information-theoretic Upper Bounds: In addition to developing efficient algorithms for particular settings, we give upper bounds on the achievable error of user-level DP model personalization via inefficient algorithms. Specifically, we consider the natural approach of using the exponential mechanism [34] to select a common structure that provides low prediction error on average across users. For the specific case of a shared linear embedding (a generalization of the linear regression setting above), when the feature vectors are drawn i.i.d. from $\mathcal{N}(0,1)^d$, and when the $\boldsymbol{v}_j^*$'s are drawn i.i.d. from $\mathcal{N}(0,1)^k$, we provide an upper bound showing that $n = \omega\left(\frac{k^{1.5} d^{1.5}}{\varepsilon}\right)$ users suffice to learn a good model, assuming m is sufficiently large for users to train the remaining parameters locally. In comparison to alternating minimization, the sample complexity is better by a factor of k.

In summary, we initiate a systematic study of differentially private model personalization in the practically important few-shot (or per-user sparse data) learning regime. We propose using users' data to learn a strong common representation/embedding using differential privacy, that can in turn be used to learn sample efficient models for each user. Using a simple but foundational problem setting, we demonstrate rigorously that this technique can indeed learn accurate common representation as well as personalized models, despite users housing only a small number of data points.

1.2 Related Work

Personalization Frameworks: Model personalization is a special case of multitask or few-shot learning [9, 24] where the goal is to leverage shared structure amongst multiple tasks to better learn the individual tasks. There are many different frameworks for multi-task learning, each capturing a different kind of shared structure. In the context of model personalization, where tasks correspond to users, two broad approaches stand out.

"Neighboring models". This approach assumes that while each user learns their own model, all or a fraction of the models are close to each other thus can be learned together [17, 24].

"Common representation". This approach, which we adopt in this paper, assumes a low-dimensional shared subspace where all points can be represented and now each user/task can learn a sample efficient model to solve the individual task [46, 37]. A common instantiation is a DNN architecture in which the weights in the last layer are user-specific but other weights are shared. Algorithmically, this second approach is more complex since it entails simultaneously finding an accurate representation of data and models building upon those representations. But several studies [37, 46] have shown it to be significantly more effective than other approaches like neighboring models.

Recent works on this approach (e.g. [43, 46, 21, 39]) follow a similar training strategy to ours—that is, they alternatively update the shared representation using gradient descent and then finetune individual classifiers [37, 29, 46]. In particular, the Almost-No-Inner-Loop (ANIL) method by [37] is most similar to the alternating optimization method that we adopt (see Algorithm 1). Theoretical understanding of these methods generally lag significantly behind their empirical success. However, several interesting recent results explain the effectiveness of these methods on simple tasks [12, 45]. Most of the papers in this domain focus on the linear regression problem with a shared low-dimensional representation that we study [45, 10, 47]. They show that one can provide much better estimates for the shared representation, and overall prediction error, by pooling information than would be possible for individual users acting alone. These existing analyses do not allow for noise in the iterations. In fact, for the general problem, the noise can lead to suboptimal solutions. Thus, a key contribution of our work is to show that in a widely studied setting, alternating minimization converges even when the minimization of $\boldsymbol{U}$ is noisy.

Privacy: In our setting, the data set is made up of users' individual data sets $D_1, ..., D_n$, where each D_j potentially contains many records (labeled training examples). Users interact via a central algorithm, which we assume for simplicity to be implemented correctly and securely (either by a trusted party or using cryptographic techniques like multiparty computation). This algorithm provides output to each of the users. We aim to control what those outputs leak about the users' input data.

That is, presence/absence of user and its entire data should not affect the outputs significantly. This notion is known as *user-level* or task-level privacy and has been widely studied in the literature [33, 30], albeit mostly without personalization component. The only works we are aware of that look at personalization (or multitask learning more generally) with user-level guarantees are [18] and [23]. Geyer et al. [18] consider the "neighboring models" approach, which cannot work in the setting we study. Jain et al. [23] consider matrix completion, which can be viewed as a version of our setting in which training examples are limited to indicator vectors (items from a known discrete set).

A few studies attempt to provide only *record-level* privacy – a significantly weaker notion of privacy where presence/absence of only single record should be undetected by the output of the model. While the notion has been studied extensively for the standard non-personalized models [26, 8], for personalized models the literature is somewhat limited [20, 31]. The work of [31] discusses both task- and record-level privacy, but ultimately provides only algorithms that satisfy the weaker guarantee. As mentioned above, our goal is to provide strong user-level privacy guarantees so such methods do not apply in our case.

1.3 Notation

We denote all matrices with bold upper case letters (e.g., $\boldsymbol{A}$), and all vectors with bold lower case letters ($\boldsymbol{a}$). Unless specified explicitly, all vectors are column vectors. We denote the clipping operation on a vector $\boldsymbol{a}$ as $\mathsf{clip}\left(\boldsymbol{a};\zeta\right) = \boldsymbol{a}\cdot\min\left\{1, \frac{\zeta}{\|\boldsymbol{a}\|_2}\right\}$.

2 Background on Privacy

Billboard model: In this paper, we operate in the billboard model [19] of differential privacy [14, 13, 35]. Consider n users, and a computing server. The server runs a differentially private algorithm on sensitive information from the users, and broadcasts the output to all the users. Each user $j \in [n]$ can then use the broadcasted output in a computation that solely relies on her data. The output of this computation is not made available to other users. A block schematic is shown in Figure 1. One important attribute of the billboard model is that it trivially satisfies joint differential privacy [27].

User-level privacy protection: In this work, we provide user-level privacy protection [15]. I.e., from the output of the algorithm available to an adversary, they will not be able to detect the presence/absence of *all the data samples belonging to a single user.* Correspondingly, in the definition of differential privacy below (Definition 2.1), a "record" consists of all the data samples belonging to a single user. Furthermore, we adhere to the replacement model of privacy, where the protection is with respect to the replacement of a user with another, instead of the presence/absence of a user.

Definition 2.1 (Differential Privacy [14, 13, 35])**.** *A randomized algorithm $\mathcal{A}$ is (ε, δ)-differentially private if for any pair of data sets D and D' that differ in one record (i.e., $|D \triangle D'| = 1$), and for all S in the output range of $\mathcal{A}$, we have*

$$\mathbf{Pr}[\mathcal{A}(D) \in S] \leq e^{\varepsilon} \cdot \mathbf{Pr}[\mathcal{A}(D') \in S] + \delta,$$

where probability is over the randomness of $\mathcal{A}$. Similarly, an algorithm $\mathcal{A}$ is (α, ρ)- Rényi differentially private (RDP) if $D_\alpha\left(\mathcal{A}(D) || \mathcal{A}(D')\right) \leq \rho$, where D_α is the Rényi divergence of order α.

3 Model Personlization via Private Alternating Minimization

In this section, we first provide a generic/meta algorithm for private model personalization (Algorithm 1 (Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$)). The main idea is to alternate between two states for T iterations, i.e., for $t \in [T]$, (i) Estimate the best embedding matrix $\boldsymbol{U}^{(t)}$ based on the current personalized models $\left[\boldsymbol{v}_1^{(t)}, \ldots, \boldsymbol{v}_n^{(t)}\right]$ while preserving *user-level* (α, ρ)-RDP, and (ii) update the personalized modes based on the updated embedding matrix $\boldsymbol{U}^{(t)}$. Finally, output $\boldsymbol{U}^{\text{priv}} \leftarrow \boldsymbol{U}^{(T+1)}$, which will be used by each user $j \in [n]$ to train her final personalized model $\boldsymbol{v}_j^{\text{priv}}$. While Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ is a fairly natural method for model personalization, to the best of our knowledge, this is the first work that formally studies the privacy/utility trade-offs under user-level privacy. Prior works [39, 36] have used similar ideas in the *non-private* meta-learning setting. The estimation of the embedding matrix can be implemented by any differentially private convex optimization algorithm (e.g., DP-SGD [41, 4, 1]). As discussed in Section 4, for specific case of linear regression, we can perturb the sufficient statistics to obtain differential privacy guarantee, and then optimize over it. A similar idea was used in [40, 38].

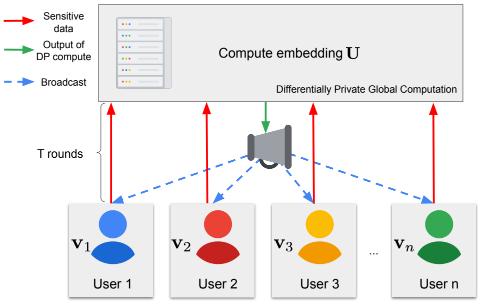

Figure 1: User-compute interaction in the billboard model. Shaded boxes represent privileged computation. $\boldsymbol{U}$ refer to the common embedding function, and $\boldsymbol{v}_j$ refers to the model for user $j \in [n]$.

We provide a formal description in Algorithm 1. In Section 4, we instantiate it in the context of personalized linear regression. There, we also provide formal excess population risk guarantees under some data generating assumption. Since Line 6 guarantees (α, ρ)-RDP, and disjoint sets of users are used in each iteration, we can conclude that the whole algorithm guarantees (α, ρ)-RDP.

4 Instantiating Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ with Linear Regression

In this section, we instantiate Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ (Algorithm 1) in the context of linear regression. While our privacy guarantees hold for any instantiation of the training data, the utility guarantees hold under the following data generating assumption.

Algorithm 1 $\mathcal{A}_{\textsf{Priv-AltMin}}$: Differentially Private Alternating Minimization Meta-algorithm

```
Require: Data sets from each user $j \in [n]$: $D_j = \{(\mathbf{x}_{ij} \in \mathbb{R}^d, y_{ij} \in \mathbb{R}) : i \in [m]\}$ for $m \bmod 4 = 0$, rank of the projector: $k$, privacy parameters: $(\alpha, \rho)$, number of iterations: $T$, initial rank-$k$ subspace matrix: $\boldsymbol{U}^{\texttt{init}}$, loss function: $\ell$.
Initialize $\boldsymbol{U}^{(1)} \leftarrow \boldsymbol{U}^{\texttt{init}}$.
Randomly permute the users $j \in [n]$ via permutation $\pi \sim_{\textsf{unif}} [n]$. Set $j \leftarrow \pi(j), \forall j \in [n]$.
for $t \in [T]$ do
    $\mathcal{S}_t \leftarrow \left[1 + \lceil \frac{(t-1)n}{T} \rceil, \lceil \frac{tn}{T} \rceil\right]$.
    Each user $j \in [\mathcal{S}_t]$ independently solves $\boldsymbol{v}_j^{(t)} \leftarrow \underset{\|\boldsymbol{v}\|_2 \leq \mathbb{R}^k}{\arg\min} \frac{4}{m} \sum_{i \in [m/4]} \ell\left(\langle (\boldsymbol{U}^{(t)})^\top \mathbf{x}_{ij}, \boldsymbol{v} \rangle; y_{ij}\right)$.
    Estimate $\boldsymbol{U}^{(t+1)} \leftarrow \underset{\boldsymbol{U} \in \mathcal{K}}{\arg\min} \frac{4}{m \cdot |\mathcal{S}_t|} \sum_{i \in [m/4+1, m/2], j \in \mathcal{S}_t} \ell\left(\langle \boldsymbol{U}^\top \mathbf{x}_{ij}, \boldsymbol{v}_j^{(t)} \rangle; y_{ij}\right)$ under $(\alpha, \rho)$-RDP, where $\mathcal{K}$ is the set of all rank-$k$ matrices with orthonormal columns in $\mathbb{R}^{d \times k}$.
end for
$\boldsymbol{U}^{\texttt{priv}} \leftarrow \boldsymbol{U}^{(T+1)}$.
```

Data generation: We instantiate the problem description in Section 1.1 as follows. There is a fixed model $\boldsymbol{v}_j^* \in \mathbb{R}^k$ for each user $j \in [n]$, and a fixed rank-k matrix with orthonormal columns $\boldsymbol{U}^* \in \mathbb{R}^{d \times k}$ across all users. Let $\boldsymbol{V}^* := [\boldsymbol{v}_1^* | \cdots | \boldsymbol{v}_n^*]$. For each feature vector $\mathbf{x}_{ij} \in \mathbb{R}^d$, the response y_{ij} is given by:

$$y_{ij} = \langle (\boldsymbol{U}^*)^\top \mathbf{x}_{ij}, \boldsymbol{v}_j^* \rangle + \boldsymbol{z}_{ij}, \;\; \boldsymbol{z}_{ij} \sim \mathcal{N}(0, \sigma_{\text{F}}^2). \tag{2}$$

In Theorem 4.2, we provide the privacy and utility guarantee for an instantiation of Algorithm 1 (Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$) where the loss function is $\ell\left(\langle \boldsymbol{U}^\top \mathbf{x}_{ij}, \boldsymbol{v} \rangle; y_{ij}\right) = \left(y_{ij} - \langle \boldsymbol{U}^\top \mathbf{x}_{ij}, \boldsymbol{v} \rangle\right)^2$. We will adhere to Assumptions 4.1 for the utility analysis.

Assumption 4.1 (Assumptions for Utility Analysis)**.** *Let $\lambda_i > 0$ be the i-th eigenvalue of $\frac{1}{n}\left(\boldsymbol{V}^* (\boldsymbol{V}^*)^\top\right)$, and let $\mu := \max_{j \in [n]} \left\|\boldsymbol{v}_j^*\right\|_2 / \sqrt{k\lambda_k}$ be the incoherence parameter. Let Noise-to-signal ratio be $\mathit{NSR} = \frac{\sigma_F}{\sqrt{\lambda_k}}$. We assume: (i) $\forall i \in [m], j \in [n], \mathbf{x}_{ij} \sim_{\textsf{iid}} \mathcal{N}(0,1)^d$, and corresponding y_{ij} be generated using (2), (ii) $m = \widetilde{\Omega}\left((1 + \mathit{NSR}) \cdot k + k^2\right)$, (iii) $n = \widetilde{\Omega}\left(\frac{\lambda_1}{\lambda_k} \cdot \mu^2 dk + d\left(\frac{\sigma_F^2}{k} + \mu^2 \lambda_k\right)^2 + \Delta_{(\varepsilon,\delta)} \cdot \left(\mathit{NSR}^2 + \mu^2 k\right) d^{3/2}\right)$. Here, $\widetilde{\Omega}(\cdot)$ hides* polylog (n, m, k).

Theorem 4.2 (Main Result. Bound on Excess Risk)**.** *Let $\boldsymbol{V}^{priv} = [\boldsymbol{v}_1^{priv}, \ldots, \boldsymbol{v}_n^{priv}]$ with*

$$\boldsymbol{v}_j^{priv} \leftarrow \underset{\boldsymbol{v} \in \mathbb{R}^k}{\arg\min} \frac{2}{m} \sum_{\frac{m}{2} < i \leq m} \left(y_{ij} - \langle (\boldsymbol{U}^{priv})^\top \mathbf{x}_{ij}, \boldsymbol{v} \rangle\right)^2.$$

Let Assumption 4.1 hold. Then, Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ with parameters in Lemma 4.4 and $\Delta_{(\varepsilon,\delta)} := \frac{\sqrt{16 \log(1/\delta)}}{\varepsilon}$ outputs $\boldsymbol{U}^{priv}$ such that i) it is (ε, δ)-differentially private, and ii) it has the following excess population risk:

$$\begin{aligned}
&\mathbb{E}\left[\mathsf{Risk}_{Pop}(\left(\boldsymbol{U}^{priv}, \boldsymbol{V}^{priv}\right); (\boldsymbol{U}^*, \boldsymbol{V}^*))\right] \leq \\
&= O\left(\frac{\Delta_{(\varepsilon,\delta)}(\sigma_F^2 + \mu^2 k^2 d \lambda_k)(\mu^4 k^3 d^2)}{n^2} + \frac{\sigma_F^2 \mu^4 k^2 d}{nm}\right) \cdot \text{polylog}\,(d, n) + \left(\frac{k}{m} + 1\right) \sigma_F^2.
\end{aligned}$$

See supplementary material for the proof.

Remark 1. Let us understand the bound above for a simple setting where the personal model for each user $\boldsymbol{v}_j^* \sim \mathcal{N}(0,1)^k$. Assuming large enough n, this implies that $\lambda_k \approx 1$ and $\mu \approx \widetilde{O}(1)$. Now even when $\boldsymbol{V}^*$ is *known a priori*, to obtain a reasonable estimate of $\boldsymbol{U}^*$, we need to solve the following linear regression problem while ensuring DP: $\boldsymbol{U}^{\texttt{priv}} = \min_{\boldsymbol{U}} \sum_{ij} (y_{ij} - \langle \mathbf{x}_{ij} (\boldsymbol{v}_j^*)^\top, \boldsymbol{U} \rangle)^2$.

Algorithm 2 Instantiating Line 6 of Algorithm 1 (Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$)

```
Require: Set of users at time step $t \in [T]$: $\mathcal{S}_t$. Current models: $\left\{\boldsymbol{v}_j^{(t)} : j \in \mathcal{S}_t\right\}$, data samples: $\{(\mathbf{x}_{ij}, y_{ij}) : j \in \mathcal{S}_t, i \in [m]\}$, privacy parameter: $\Delta_{(\varepsilon,\delta)}$, clipping threshold for model: $\eta$, clipping threshold for response: $\zeta$.
$\boldsymbol{W}_{ij} = \textsf{clip}\left(\overrightarrow{\mathbf{x}_{ij}\boldsymbol{v}_j^\top}; \eta\right)$ and $\widetilde{y}_{ij} = \textsf{clip}\left(y_{ij}; \zeta\right)$ for all $i \in [m/4+1, m/2]$, $j \in \mathcal{S}_t$.
$\boldsymbol{W}_{\texttt{priv}} \leftarrow \sum_{j \in \mathcal{S}_t, i \in [m/4+1, m/2]} \boldsymbol{W}_{ij}\boldsymbol{W}_{ij}^\top + \mathcal{N}_{\textsf{sym}}\left(0, m^2\eta^4\Delta_{(\varepsilon,\delta)}^2/4\right)^{dk \times dk}$, and
$\boldsymbol{b}_{\texttt{priv}} \leftarrow \sum_{j \in \mathcal{S}_t, i \in [m/4+1, m/2]} \widetilde{y}_{ij}\boldsymbol{W}_{ij} + \mathcal{N}\left(0, m^2\zeta^2\eta^2\Delta_{(\varepsilon,\delta)}^2/4\right)^{dk}$
$\vec{\boldsymbol{Z}}^{(t+1)} \leftarrow \arg\min_{\boldsymbol{u} \in \mathbb{R}^{dk}} \frac{4}{m \cdot |\mathcal{S}_t|}\left(\boldsymbol{u}^\top \boldsymbol{W}_{\texttt{priv}}\boldsymbol{u} - 2\boldsymbol{u}^\top \boldsymbol{b}_{\texttt{priv}}\right)$
return $\boldsymbol{U}^{(t+1)} \leftarrow Q$ part of the $QR$-decomposition of $\boldsymbol{Z}^{(t+1)}$
```

Note that $\mathbf{x}_{ij}(\boldsymbol{v}_j^*)^\top$ is isotropic. Now, *without* differential privacy, the information theoretical optimal estimation error is $\Theta\left(\sigma_{\mathrm{F}}^2 \cdot \frac{dk}{nm}\right)$, where dk is the size of the linear regression problem and mn is the number of samples. Now, if we were to solve the above regression problem with DP, the best known algorithm [40] will have an additional error of $\widetilde{O}\left(\left(\kappa \cdot \frac{dk}{n\varepsilon}\right)^2\right)$, where $\kappa = \sigma_{\mathrm{F}} + \max_{ij} \|\mathbf{x}_{ij}(\boldsymbol{v}_j^*)^\top\|_F \cdot \|\boldsymbol{U}^*\|_F = \widetilde{O}(\sigma_{\mathrm{F}} + \sqrt{dk^2})$. Note that the first two terms in Theorem 4.2 indeed match $O\left(\left(\kappa \cdot \frac{dk}{n\varepsilon}\right)^2 + \sigma_{\mathrm{F}}^2 \cdot \frac{dk}{nm}\right)$ up to an additional factor of k and up to $\mathrm{polylog}\,(d, n)$ factors. Finally, the last error term in the above theorem is due to excess risk in estimating $\boldsymbol{v}^*$ for a given user with m samples, and is information theoretically optimal.

Remark 2. Under the assumption in Remark 1 and for $\sigma_{\mathrm{F}} = 0$, the sample complexity for Theorem 4.2 is $n = \widetilde{\omega}(k^{2.5}d^{1.5}/\varepsilon + d)$ and $m = \widetilde{\omega}(k^2)$. Note that, for $\varepsilon \to \infty$, the complexity is $O(k)$ worse than the information theoretic optimal. Furthermore, the sample complexity suffers from an additional $\sqrt{d}$ for constant ε compared to non-private case. Even for standard linear regression, a similar additional $\sqrt{d}$ factor is present in the sample complexity bound [40]; we leave further investigation into the optimal sample complexity for future work.

In Section 4.1, we show an instantiation of Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ (Algorithm 1) s.t. if the embedding matrix ($\boldsymbol{U}^{\texttt{init}}$) is initialized well, then $\boldsymbol{U}^{\texttt{priv}}\left(\boldsymbol{U}^{\texttt{priv}}\right)^\top$ converges in $\|\cdot\|_F$ to $\boldsymbol{U}^*(\boldsymbol{U}^*)^\top$. In Section 4.2, we provide an algorithm to obtain a good initialization of the embedding matrix ($\boldsymbol{U}^{\texttt{init}}$). Combining these two results imply Theorem 4.2.

4.1 Local Subspace Convergence

In Algorithm 2, we instantiate Line 6 of Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$. For any matrix $\boldsymbol{A} \in \mathbb{R}^{d_1 \times d_2}$, let $\vec{\boldsymbol{A}} \in \mathbb{R}^{d_1 d_2}$ be the vectorized representation with columns of $\boldsymbol{A}$ placed consecutively. Let $\mathcal{N}_{\textsf{sym}}(0, \sigma^2)^{d \times d}$ denote a Wigner matrix with entries drawn i.i.d. from $\mathcal{N}(0, \sigma^2)$. The privacy guarantee of Algorithm 2 is presented in Lemma 4.3 and the local subspace guarantee in Lemma 4.4.

Lemma 4.3 (Privacy guarantee)**.** *If we set $\Delta_{(\varepsilon,\delta)} = \sqrt{8\log(1/\delta)}/\varepsilon$, then instantiation of Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$ with Algorithm 2 is (ε, δ)-differentially private in the billboard model.*

Lemma 4.4 (Local Subspace Convergence)**.** *Recall Assumptions 4.1. In Algorithm 2, let model clipping threshold $\eta = \widetilde{O}(\mu\sqrt{\lambda_k dk})$, and response clipping threshold $\zeta = \widetilde{O}\left(\sigma_F + \mu\sqrt{k\lambda_k}\right)$. Let the number of iterations of Algorithm 1 (Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$) be $T = \Omega\left(\log\left(\frac{(\lambda_1/\lambda_k)}{NSR + \Delta_{(\varepsilon,\delta)}}\right)\right)$. Finally, assume $\boldsymbol{U}^{init}$ be s.t. $\|(\mathbb{I} - \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top)\boldsymbol{U}^{init}\|_F \leq \frac{\lambda_k}{32\lambda_1}$. We have the following for Algorithm 1 (Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$), instantiated with Algorithm 2, w.p. at least $1 - 1/n^{10}$ (over the randomness of data generation and the algorithm):*

$$\left\|\left(\mathbb{I} - \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top\right)\boldsymbol{U}^{priv}\right\|_F = \widetilde{O}\left(\frac{\Delta_{(\varepsilon,\delta)}(NSR + \mu\sqrt{dk^2})\mu\sqrt{k^2d^2}}{n} + \frac{NSR \cdot \mu\sqrt{kd}}{\sqrt{nm}}\right).$$

Here, the noise-to-signal-ratio $\mathtt{NSR} = \frac{\sigma_F}{\sqrt{\lambda_k}}$ *and privacy parameter* $\Delta_{(\varepsilon,\delta)} = \frac{\sqrt{8\log(1/\delta)}}{\varepsilon}$. *In* $\widetilde{O}(\cdot)$, *we hide* $\mathrm{polylog}\,(d, n)$.

See supplementary material for the proofs. The analysis of Lemma 4.4 roughly follows the analysis of alternating minimization [45], while accounting for the noise introduced due to privacy. At each iteration, we show that the embedding subspace gets closer in the Frobenius norm, and each of the personalized models gets closer in the ℓ_2-norm.

4.2 Initialization Algorithm

In Algorithm 3, we describe a private estimator for the estimation of $\boldsymbol{U}^*$. This estimator eventually gets used in initializing the linear regression instantiation of Algorithm 1. We provide the privacy and subspace closeness guarantees in Lemma 4.5 and 4.6, with proofs in supplementary material.

Algorithm 3 $\mathcal{A}_{\textsf{Priv-init}}$: Private Initialization Algorithm for Algorithm $\mathcal{A}_{\textsf{Priv-AltMin}}$

```
Require: Data sets from each user $j \in [n]$: $D_j = \{(\mathbf{x}_{ij} \in \mathbb{R}^d, y_{ij} \in \mathbb{R}) : i \in [m]\}$, clipping bound
    for response: $\zeta$, noise standard dev. for privacy: $\Delta_{(\varepsilon,\delta)}$, and rank of the orthonormal basis: $k$.
  $\boldsymbol{W}_{ij} \leftarrow \mathsf{sym}\left(\frac{\mathbf{x}_{(2i)j}\mathbf{x}_{(2i+1)j}^\top}{\|\mathbf{x}_{(2i)j}\|_2 \cdot \|\mathbf{x}_{(2i+1)j}\|_2} \cdot \mathsf{clip}\left(y_{(2i)j}; \zeta\right) \cdot \mathsf{clip}\left(y_{(2i+1)j}; \zeta\right)\right)$ for all $i \in [m/2]$ and $j \in [n]$. Here, $\mathsf{sym}(\boldsymbol{W})$ makes a matrix $\in \mathbb{R}^{d\times d}$ symmetric by replicating the upper triangle.
  $\boldsymbol{M}^{\mathtt{Noisy}} \leftarrow \frac{2}{nm}\left(\sum_{i\in[m/2], j\in[n]} \boldsymbol{W}_{ij} + \mathcal{N}_{\mathsf{sym}}\left(0, \Delta_{(\varepsilon,\delta)}^2 \zeta^4 m^2\right)^{d\times d}\right)$.
  $\boldsymbol{U}^{\mathtt{priv}} \leftarrow$ Top-$k$ eigenvectors of $\boldsymbol{M}^{\mathtt{Noisy}}$ as columns.
```

Lemma 4.5 (Privacy guarantee)**.** *If we set* $\Delta_{(\varepsilon,\delta)} = \sqrt{8\log(1/\delta)}/\varepsilon$, *Algorithm 3 (Algorithm* $\mathcal{A}_{\textsf{Priv-init}}$*) is* (ε, δ)*-differentially private.*

Lemma 4.6 (Subspace closeness)**.** *Recall Assumptions 4.1. Let the clipping bound for response be* $\zeta = \widetilde{O}(\sigma_F + \mu\sqrt{k\lambda_k})$. *We have the following for Algorithm 3 (Algorithm* $\mathcal{A}_{\textsf{Priv-init}}$*) w.p. at least* $1 - 1/n^{10}$*:*

$$\left\|\left(\mathbb{I} - \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top\right)\boldsymbol{U}^{\mathtt{priv}}\right\|_2 = \widetilde{O}\left(\frac{\Delta_{(\varepsilon,\delta)}\left(\mathtt{NSR}^2 + \mu^2 k\right) d^{3/2}}{n} + \frac{(\mathtt{NSR}^2 + \mu^2 k)\sqrt{d}}{\sqrt{nm}}\right).$$

Here, privacy parameter $\Delta_{(\varepsilon,\delta)} = \frac{\sqrt{8\log(1/\delta)}}{\varepsilon}$. *In* $\widetilde{O}(\cdot)$, *we hide* $\mathrm{polylog}\,(d, n)$.

The proof goes via direct analysis of the distance between the estimated subspace from the training examples, and the true subspace. While the convergence guarantee in Lemma 4.6 is unconditional, it is weaker than Lemma 4.4, especially in its dependence on k and $\mathtt{NSR}$.

Lemma 4.6 implies that under Assumption 4.1, $\left\|\left(\mathbb{I} - \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top\right)\boldsymbol{U}^{\mathtt{priv}}\right\|_F = O\left(\frac{\lambda_1}{\lambda_k}\right)$, which is sufficient to satisfy the initialization condition in Lemma 4.4. Hence, if we initialize $\boldsymbol{U}$ using Algorithm 3 (Algorithm $\mathcal{A}_{\textsf{Priv-init}}$) with a *disjoint* set of samples for each user, it immediately follows that the the local convergence guarantee in Lemma 4.4 is indeed a global convergence guarantee.

5 Exponential Mechanism based Model Personalization

In this section, we take a more general approach towards outputting a projector $\boldsymbol{U}^{\mathtt{priv}}$ that approximately minimizes the excess population risk without worrying about actually estimating the projector onto $\boldsymbol{U}^*$. Here, as we only care about low-excess risk, as opposed to subspace closeness, we can guarantee better convergence under milder assumptions. Recall the loss function $\mathcal{L}_{\mathtt{Pop}}(\boldsymbol{U}, \boldsymbol{V})$ from (1). We want to optimize $\min\limits_{\boldsymbol{U}\in\mathcal{K}}\left(\min\limits_{\boldsymbol{V}\in\mathbb{R}^{d\times n}, \|\boldsymbol{v}_j\|_2 \le C} \mathcal{L}_{\mathtt{Pop}}(\boldsymbol{U}, \boldsymbol{V})\right)$ while ensuring ε-DP in the billboard model. (Here $\mathcal{K} \in \mathbb{R}^{d\times k}$ is the set of matrices with orthonormal columns, and $\boldsymbol{v}_j$ corresponds to the j-th column of $\boldsymbol{V}$.) To that end, we will use the exponential mechanism [34], over an ℓ_F-net of radius ϕ over $\mathcal{K}$. The algorithm is presented in Algorithm 4 (Algorithm $\mathcal{A}_{\textsf{Exp}}$).

Algorithm 4 $\mathcal{A}_{\mathsf{Exp}}$: Joint Differentially Private ERM via Exponential Mechanism

```
Require: Data sets from each user $j \in [n]$: $D_j = \{(\mathbf{x}_{ij} \in \mathbb{R}^d, y_{ij} \in \mathbb{R}) : i \in [m]\}$ where $m \mod 2 = 0$, model $\ell_2$-norm constraint: $C$, clipping bound on the projected features: $L_f$ (see Theorem 5.2 below), privacy parameter: $\varepsilon$, and rank of the orthonormal basis: $k$, net width: $\phi$, loss function: $\ell : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ with $\xi$ being the Lipschitz constant of the first parameter.
Define a score function for any rank-$k$ matrix with orthonormal columns $\boldsymbol{U} \in \mathbb{R}^{d \times k}$ as
    $\mathsf{score}(\boldsymbol{U}) = \sum_{j \in [n]} \left( \min_{\|\boldsymbol{v}_j\|_2 \le C} \frac{2}{m} \sum_{i \in [m/2]} \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_j \rangle; y_{ij}\right) \right)$.
Define a net $\mathcal{N}^\phi$ of $\|\cdot\|_F$-radius $\phi$ over matrices with orthonormal columns in $\mathbb{R}^{d \times k}$.
Sample $\boldsymbol{U}^{\mathtt{priv}} \in \mathcal{N}^\phi$ with $\mathbf{Pr}[\boldsymbol{U}^{\mathtt{priv}} = \boldsymbol{U}] \propto \exp\left(-\frac{\varepsilon n}{8 L_f C \xi} \cdot \mathsf{score}(\boldsymbol{U})\right)$.
Each user $j \in [n]$ independently estimates $\boldsymbol{v}_j^{\mathtt{priv}} \leftarrow \arg\min_{\|\boldsymbol{v}\|_2 \le C} \frac{2}{m} \sum_{i=m/2+1}^{m} \ell\left(\langle \left(\boldsymbol{U}^{\mathtt{priv}}\right)^\top \mathbf{x}_{ij}, \boldsymbol{v} \rangle; y_{ij}\right)$.
```

The privacy analysis of Algorithm $\mathcal{A}_{\mathsf{Exp}}$ follows from the standard analysis of exponential mechanism, and the utility analysis goes via first proving an excess empirical risk bound, and then appealing to uniform convergence to get to excess population risk bound.

Theorem 5.1 (Privacy guarantee)**.** *Algorithm 4 is ε-differentially private in the billboard model.*

Theorem 5.2 (Utility guarantee)**.** *Suppose the loss function ℓ is ξ-Lipschitz in the first parameter, and C is the bound on the constraint set. Set the net size $\phi = 1/(\varepsilon n)$ in Algorithm 4. Assuming that the feature vectors are drawn i.i.d. from $\mathcal{N}(0,1)^d$, and setting $L_f = 40\sqrt{d} \cdot \log(nm)$, we have*

$$\mathbb{E}\left[\mathsf{Risk}_{Pop}(\left(\boldsymbol{U}^{priv}, \boldsymbol{V}^{priv}\right); (\boldsymbol{U}^*, \boldsymbol{V}^*))\right] = O\left(\xi C \cdot \left(\frac{\sqrt{k^2 d^3}}{\varepsilon n} + \frac{\sqrt{d}}{\sqrt{nm}} + \frac{\sqrt{k}}{\sqrt{m}}\right)\right) \cdot \mathrm{polylog}\,(d, n)\,.$$

Here, $\boldsymbol{U}^*$ and $\boldsymbol{V}^*$ are any fixed parameters from the corresponding domains.

See supplementary material for the proofs of Theorems 5.1 and 5.2.

Comparison of the utility guarantee to Theorem 4.2: The utility guarantee for Algorithm $\mathcal{A}_{\mathsf{Exp}}$ (Theorem 5.2) is much more general than that in Theorem 4.2. Unlike Theorem 4.2, it allows arbitrary Lipschitz loss function ℓ, and any distribution over the feature vectors. However, for linear regression with i.i.d. spherical normal feature vectors and setting the diameter of the constraint set $C = \sqrt{k}$, one can make Theorems 4.2 and 5.2 comparable. Theorem 4.2 shows an excess population risk $\widetilde{O}\left(\frac{k^5 d^3}{\varepsilon^2 n^2} + \frac{k}{m}\right)$ whereas Theorem 5.2 gives $\widetilde{O}\left(\frac{\sqrt{k^3 d^3}}{\varepsilon n} + \sqrt{\frac{k^2}{m}}\right)$. Theorem 4.2 is tighter in the regime where $n = \Omega(k^{3.5} d^{1.5}/\varepsilon)$. This difference is comparable to the so-called *fast rates* [42]. However, the sample complexity of Theorem 5.2 is better in terms of m by a factor of $k^{1.5}$.

6 Conclusion

In this paper we studied the problem of personalized supervised learning with user-level differential privacy. Through our framework and Algorithm 1, we demonstrated that we can indeed learn accurate shared *linear* representation of the data, despite a limited number of samples-per-user and while preserving each user's privacy. Our error bounds and sample complexity bounds are nearly optimal in key parameters and are in fact, comparable to the best known bounds available for a much simpler linear regression problem.

Limitations and Future Directions: This work leads to several interesting questions: (i) In our model, can we provide similar privacy/utility trade-offs for deep networks based embedding functions instead of a linear embedding function, (ii) Can we make a variant of the exponential mechanism algorithm computationally feasible?, and (iii) Empirically validate the privacy/utility trade-offs on real world data sets.

Broader Impact: As more and more ML models are personalized for user tastes, ensuring privacy of individuals' data is paramount to a fair, responsible system. We provide a rigorous framework to design such solutions, which hopefully will motivate practitioners and researchers to make privacy as a first class citizen while designing their personalization based ML system.

References

[1] Martín Abadi, Andy Chu, Ian J. Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proc. of the 2016 ACM SIGSAC Conf. on Computer and Communications Security (CCS'16)*, pages 308–318, 2016.

[2] Peter L. Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *J. Mach. Learn. Res.*, 2003.

[3] Raef Bassily, Vitaly Feldman, Kunal Talwar, and Abhradeep Guha Thakurta. Private stochastic convex optimization with optimal rates. In *NeurIPS*, pages 11279–11288, 2019.

[4] Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *Proc. of the 2014 IEEE 55th Annual Symp. on Foundations of Computer Science (FOCS)*, 2014.

[5] Emmanuel J. Candès and Yaniv Plan. Tight oracle bounds for low-rank matrix recovery from a minimal number of random measurements. *CoRR*, abs/1001.0339, 2010.

[6] Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 267–284, 2019.

[7] Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, et al. Extracting training data from large language models. *arXiv preprint arXiv:2012.07805*, 2020.

[8] Kamalika Chaudhuri, Claire Monteleoni, and Anand D Sarwate. Differentially private empirical risk minimization. *Journal of Machine Learning Research*, 12(Mar):1069–1109, 2011.

[9] Yinbo Chen, Xiaolong Wang, Zhuang Liu, Huijuan Xu, and Trevor Darrell. A new meta-baseline for few-shot learning. *arXiv preprint arXiv:2003.04390*, 2020.

[10] Liam Collins, Hamed Hassani, Aryan Mokhtari, and Sanjay Shakkottai. Exploiting shared representations for personalized federated learning. *arXiv preprint arXiv:2102.07078*, 2021.

[11] Chandler Davis. The rotation of eigenvectors by a perturbation. *Journal of Mathematical Analysis and Applications*, 6(2):159–173, 1963.

[12] Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

[13] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology—EUROCRYPT*, pages 486–503, 2006.

[14] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Proc. of the Third Conf. on Theory of Cryptography (TCC)*, pages 265–284, 2006.

[15] Cynthia Dwork, Moni Naor, Toniann Pitassi, and Guy N. Rothblum. Differential privacy under continual observation. In *STOC*, 2010.

[16] Cynthia Dwork, Kunal Talwar, Abhradeep Thakurta, and Li Zhang. Analyze gauss: optimal bounds for privacy-preserving principal component analysis. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pages 11–20, 2014.

[17] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. *arXiv preprint arXiv:1703.03400*, 2017.

[18] Robin C. Geyer, Tassilo Klein, and Moin Nabi. Differentially private federated learning: A client level perspective. *CoRR*, abs/1712.07557, 2017.

[19] Justin Hsu, Zhiyi Huang, Aaron Roth, Tim Roughgarden, and Zhiwei Steven Wu. Private matchings and allocations. In *STOC*, 2014.

[20] Rui Hu, Yuanxiong Guo, Hongning Li, Qingqi Pei, and Yanmin Gong. Personalized federated learning with differential privacy. *IEEE Internet Things J.*, 7(10):9530–9539, 2020.

[21] Shaoli Huang and Dacheng Tao. All you need is a good representation: A multi-level and classifier-centric representation for few-shot learning. *arXiv preprint arXiv:1911.12476*, 2019.

[22] Prateek Jain, Praneeth Netrapalli, and Sujay Sanghavi. Low-rank matrix completion using alternating minimization. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 665–674, 2013.

[23] Prateek Jain, Om Dipakbhai Thakkar, and Abhradeep Thakurta. Differentially private matrix completion revisited. In *International Conference on Machine Learning*, pages 2215–2224. PMLR, 2018.

[24] Muhammad Abdullah Jamal and Guo-Jun Qi. Task agnostic meta-learning for few-shot learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11719–11727, 2019.

[25] Yihan Jiang, Jakub Kone$v$cný, Keith Rush, and Sreeram Kannan. Improving federated learning personalization via model agnostic meta learning. *CoRR*, abs/1909.12488, 2019.

[26] Shiva Prasad Kasiviswanathan, Homin K. Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam D. Smith. What can we learn privately? In *49th Annual IEEE Symp. on Foundations of Computer Science (FOCS)*, pages 531–540, 2008.

[27] Michael Kearns, Mallesh Pai, Aaron Roth, and Jonathan Ullman. Mechanism design in large games: Incentives and privacy. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 403–410, 2014.

[28] Yehuda Koren, Robert Bell, and Chris Volinsky. Matrix factorization techniques for recommender systems. *Computer*, 42(8):30–37, 2009.

[29] Kwonjoon Lee, Subhransu Maji, Avinash Ravichandran, and Stefano Soatto. Meta-learning with differentiable convex optimization. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10649–10657. IEEE Computer Society, 2019.

[30] Daniel Levy, Ziteng Sun, Kareem Amin, Satyen Kale, Alex Kulesza, Mehryar Mohri, and Ananda Theertha Suresh. Learning with user-level privacy. *CoRR*, abs/2102.11845, 2021.

[31] Jeffrey Li, Mikhail Khodak, Sebastian Caldas, and Ameet Talwalkar. Differentially private meta-learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[32] Percy Liang. Cs229t/stat231: Statistical learning theory (winter 2016), 2016.

[33] H. Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

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

[35] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275. IEEE, 2017.

[36] Seewong Oh, Prateek Jain, and Kiran Thekumparampil. Sample efficient linear meta-learning by alternating minimization. *Personal communication*, 2021.

[37] Aniruddh Raghu, Maithra Raghu, Samy Bengio, and Oriol Vinyals. Rapid learning or feature reuse? towards understanding the effectiveness of MAML. *arXiv preprint arXiv:1909.09157*, 2019.

[38] Or Sheffet. Old techniques in differentially private linear regression. In *ALT*, 2019.

[39] Karan Singhal, Hakim Sidahmed, Zachary Garrett, Shanshan Wu, Keith Rush, and Sushant Prakash. Federated reconstruction: Partially local federated learning. *CoRR*, abs/2102.03448, 2021.

[40] Adam Smith, Abhradeep Thakurta, and Jalaj Upadhyay. Is interaction necessary for distributed private learning? In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 58–77. IEEE, 2017.

[41] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE Global Conference on Signal and Information Processing*, pages 245–248. IEEE, 2013.

[42] Karthik Sridharan, Shai Shalev-shwartz, and Nathan Srebro. Fast rates for regularized objectives. In *Advances in Neural Information Processing Systems*. Curran Associates, Inc., 2009.

[43] Chen Sun, Abhinav Shrivastava, Saurabh Singh, and Abhinav Gupta. Revisiting unreasonable effectiveness of data in deep learning era. In *Proceedings of the IEEE international conference on computer vision*, pages 843–852, 2017.

[44] Terence Tao. *Topics in random matrix theory*, volume 132. American Mathematical Soc., 2012.

[45] Kiran Koshy Thekumparampil, Prateek Jain, Praneeth Netrapalli, and Sewoong Oh. Sample efficient linear meta-learning by alternating minimization, 2021.

[46] Yonglong Tian, Yue Wang, Dilip Krishnan, Joshua B Tenenbaum, and Phillip Isola. Rethinking few-shot image classification: a good embedding is all you need? *arXiv preprint arXiv:2003.11539*, 2020.

[47] Nilesh Tripuraneni, Chi Jin, and Michael I Jordan. Provable meta-learning of linear representations. *arXiv preprint arXiv:2002.11684*, 2020.

[48] Joel A Tropp. User-friendly tail bounds for sums of random matrices. *Foundations of computational mathematics*, 2012.

[49] Joel A Tropp. An introduction to matrix concentration inequalities. *arXiv preprint arXiv:1501.01571*, 2015.

[50] Kangkang Wang, Rajiv Mathews, Chloé Kiddon, Hubert Eichner, Françoise Beaufays, and Daniel Ramage. Federated evaluation of on-device personalization. *CoRR*, abs/1910.10252, 2019.

[51] Tao Yu, Eugene Bagdasaryan, and Vitaly Shmatikov. Salvaging federated learning by local adaptation. *CoRR*, abs/2002.04758, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]

   (b) Did you describe the limitations of your work? [Yes] Described in Section 6.

   (c) Did you discuss any potential negative societal impacts of your work? [N/A] We present a theoretical and rigorous study of model personalization with privacy as a first class citizen. See Section 6.

   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] Described in Assumption 4.1

   (b) Did you include complete proofs of all theoretical results? [Yes] In supplementary material.

3. If you ran experiments...[N/A]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...[N/A]

5. If you used crowdsourcing or conducted research with human subjects...[N/A]

A Missing Proofs from Section 4

A.1 Proof of Lemma 4.3

Proof. We will show that $\boldsymbol{W}_{\texttt{priv}}$ and $\boldsymbol{b}_{\texttt{priv}}$ in Algorithm 2 guarantee differential privacy. As the $\arg\min$ can be computed given the two quantities, it will guarantee differential privacy by sequential composition.

For any j, denote $\boldsymbol{A}_j = \sum_{i\in[m/4+1,m/2]} \boldsymbol{W}_{ij}\boldsymbol{W}_{ij}^\top$ and $\boldsymbol{b}_j = \sum_{i\in[m/4+1,m/2]} \widetilde{y}_{ij}\boldsymbol{W}_{ij}$. For any iteration t, let $\boldsymbol{A} = \sum_{j\in\mathcal{S}_t} \boldsymbol{A}_j$ and $\boldsymbol{b} = \sum_{j\in\mathcal{S}_t} \boldsymbol{b}_j$. Considering neighboring datasets D and D' such that user j's data in D is replaced by user j^*'s. If $j \notin \mathcal{S}_t$ in iteration t, $\boldsymbol{A}$ and $\boldsymbol{b}$ will be the same. Otherwise, A would change by $\Delta\boldsymbol{A} = \boldsymbol{A}_{j^*} - \boldsymbol{A}_j$ and $\boldsymbol{b}$ by $\Delta\boldsymbol{b} = \boldsymbol{b}_{j^*} - \boldsymbol{b}_j$. We will bound the two quantities.

- For $\Delta\boldsymbol{A}$: According to the definitions, we have $\|\boldsymbol{W}_{ij}\|_2 \le \eta$. Consider the Frobenius norm of matrix $\boldsymbol{W}_{ij}\boldsymbol{W}_{ij}^\top$. For any vector x, we have $\|\mathbf{x}\mathbf{x}^\top\|_F = \sqrt{\sum_{p,q} x_p^2 x_q^2} = \sqrt{\sum_p x_p^2 \sum_q x_q^2} = \|\mathbf{x}\|_2^2$. Therefore, we have $\left\|\boldsymbol{W}_{ij}\boldsymbol{W}_{ij}^\top\right\|_F = \|\boldsymbol{W}_{ij}\|_2^2 \le \eta^2$, and thus $\|\boldsymbol{A}_j\|_F \le m\eta^2/4$, and $\|\Delta\boldsymbol{A}\|_F \le \|\boldsymbol{A}_j\|_F + \|\boldsymbol{A}_{j^*}\|_F \le m\eta^2/2$.
- For $\Delta\boldsymbol{b}$: Again according to definition, we have $|\widetilde{y}_{ij}| \le \zeta$ for any j. Thus $\|\boldsymbol{b}_j\|_2 \le m\eta\zeta/4$ for any j, and $\|\Delta\boldsymbol{b}\|_2 \le m\eta\zeta/2$.

Applying Gaussian mechanism, adding noise $\mathcal{N}(0, m^2\eta^2\zeta^2\Delta^2_{(\varepsilon,\delta)}/4)^{dk}$ to $\boldsymbol{b}$ guarantees $(\alpha, \alpha/(2\Delta^2_{(\varepsilon,\delta)}))$-RDP. As for $\boldsymbol{A}$, adding $\mathcal{N}(0, m^2\eta^4\Delta^2_{(\varepsilon,\delta)}/4)^{dk\times dk}$ to the vectorized version of $\boldsymbol{A}$ guarantees $(\alpha, \alpha/(2\Delta^2_{(\varepsilon,\delta)}))$-RDP. We can reshape the vectorized $\boldsymbol{A}$ to get the matrix version, which is a postprocessing step and does not affect the privacy guarantee. Notice that $\boldsymbol{A}$ is a symmetric matrix. We can thus copy its upper triangle to the lower, which is equivalent to adding a symmetric Gaussian matrix to $\boldsymbol{A}$ as stated in the algorithm.

By sequential composition, one run of Algorithm 2 guarantees $(\alpha, \alpha/\Delta^2_{(\varepsilon,\delta)})$-RDP. Notice that Algorithm 1 calls Algorithm 2 for T times on disjoint sets of users. So by parallel composition, Algorithm 1 guarantees $(\alpha, \alpha/\Delta^2_{(\varepsilon,\delta)})$-RDP, which translates to $\left(\frac{\alpha}{\Delta^2_{(\varepsilon,\delta)}} + \frac{\log(1/\delta)}{\alpha-1}, \delta\right)$-DP for any ε, δ by standard conversion from RDP to approximate DP. Optimizing over α, we get $\left(\frac{1}{\Delta^2_{(\varepsilon,\delta)}} + \frac{2\sqrt{\log(1/\delta)}}{\Delta_{(\varepsilon,\delta)}}, \delta\right)$-DP. Solving $\Delta_{(\varepsilon,\delta)}$ from $\frac{1}{\Delta^2_{(\varepsilon,\delta)}} + \frac{2\sqrt{\log(1/\delta)}}{\Delta_{(\varepsilon,\delta)}} \le \varepsilon$, we have $\Delta_{(\varepsilon,\delta)} \ge \frac{\sqrt{\log(1/\delta)}+\sqrt{\log(1/\delta)+\varepsilon}}{\varepsilon}$. Therefore, if $\varepsilon \le \log(1/\delta)$, it suffices to guarantee (ε,δ)-DP by setting $\Delta_{(\varepsilon,\delta)} = \frac{\sqrt{8\log(1/\delta)}}{\varepsilon}$. □

A.2 Proof of Lemma 4.5

Proof. We will show that publishing $\boldsymbol{M}^{\texttt{Noisy}}$ guarantees differential privacy. As $\boldsymbol{W}_{ij}$'s and $\boldsymbol{M}^{\texttt{Noisy}}$ are all symmetric, for privacy analysis, it suffices to consider the upper triangles of them. Let $\mathsf{up}(X)$ denote the upper triangle of matrix X flatten into a vector. Let $\boldsymbol{w}_{ij} = \mathsf{up}(\boldsymbol{W}_{ij})$, $\boldsymbol{w} = \sum_{i,j} \boldsymbol{w}_{ij}$, and $\widetilde{\boldsymbol{w}} = \sum_{i,j} \boldsymbol{w}_{ij} + \mathsf{up}\left(\mathcal{N}_{\mathsf{sym}}\left(0, \Delta^2_{(\varepsilon,\delta)}\zeta^4 m^2\right)^{d^2}\right)$. It is easy to see that $\boldsymbol{M}^{\texttt{Noisy}}$ can be formed by postprocessing $\widetilde{\boldsymbol{w}}$. We will thus prove the privacy property of $\widetilde{\boldsymbol{w}}$, which directly translate to the privacy guarantee of $\boldsymbol{M}^{\texttt{Noisy}}$.

Consider neighboring datasets D and D' such that user j's data in D is replaced by user j^*'s data in D'. Then the corresponding $\boldsymbol{w}$ would differ by $\sum_i \boldsymbol{w}_{ij^*} - \sum_i \boldsymbol{w}_{ij}$. We will analyze its ℓ_2 norm. For any i and j, we have

$$\begin{aligned}
&\left\|\frac{\mathbf{x}_{(2i)j}\mathbf{x}_{(2i+1)j}^\top}{\left\|\mathbf{x}_{(2i)j}\right\|_2 \cdot \left\|\mathbf{x}_{(2i+1)j}\right\|_2} \cdot \mathsf{clip}\left(y_{(2i)j};\zeta\right) \cdot \mathsf{clip}\left(y_{(2i+1)j};\zeta\right)\right\|_F \\
\le&\zeta^2 \frac{\left\|\mathbf{x}_{(2i)j}\mathbf{x}_{(2i+1)j}^\top\right\|_F}{\left\|\mathbf{x}_{(2i)j}\right\|_2 \cdot \left\|\mathbf{x}_{(2i+1)j}\right\|_2} = \zeta^2.
\end{aligned} \tag{3}$$

where $\|\cdot\|_F$ denotes the Frobenius norm. The inequality follows from the definition of the clipping operation, and the equality follows because for two vectors a, b, we have $\left\|ab^\top\right\|_F^2 = \sum_{p,q}(a_p b_q)^2 = \sum_p a_p^2 \cdot \sum_q b_q^2 = \|a\|_2^2 \|b\|_2^2$. Therefore, we have $\|\boldsymbol{w}_{ij}\|_2 \leq \zeta^2$ for any i, j, which implies $\|\sum_i \boldsymbol{w}_{ij^*} - \sum_i \boldsymbol{w}_{ij}\|_2 \leq \sum_i \|\boldsymbol{w}_{ij^*}\|_2 + \sum_i \|\boldsymbol{w}_{ij}\|_2 \leq m\zeta^2$ for any j, i.e., the ℓ_2 sensitivity of $\boldsymbol{w}$ is $m\zeta^2$.

Using Gaussian mechanism, adding noise $\mathcal{N}(0, m^2\zeta^4\Delta^2_{(\varepsilon,\delta)}\mathbb{I})$ to $\boldsymbol{w}$ guarantees $(\alpha, \alpha/(2\Delta^2_{(\varepsilon,\delta)}))$-RDP for any order $\alpha \geq 1$, which translates to $\left(\frac{\alpha}{2\Delta^2_{(\varepsilon,\delta)}} + \frac{\log(1/\delta)}{\alpha-1}, \delta\right)$-DP for any $\varepsilon, \delta > 0$. Optimizing over α, it translates to $\left(\frac{1}{2\Delta^2_{(\varepsilon,\delta)}} + \frac{\sqrt{2\log(1/\delta)}}{\Delta_{(\varepsilon,\delta)}}, \delta\right)$-DP. Solving $\frac{1}{2\Delta^2_{(\varepsilon,\delta)}} + \frac{\sqrt{2\log(1/\delta)}}{\Delta_{(\varepsilon,\delta)}} \leq \varepsilon$, we get $\Delta_{(\varepsilon,\delta)} \geq \frac{\sqrt{\log(1/\delta)}+\sqrt{\log(1/\delta)+\varepsilon}}{\sqrt{2}\varepsilon}$. Therefore, if $\varepsilon \leq \log(1/\delta)$, it suffices to guarantee (ε, δ)-DP by setting $\Delta_{(\varepsilon,\delta)} = \frac{\sqrt{8\log(1/\delta)}}{\varepsilon}$. □

A.3 Proof of Lemma 4.6

Proof. Let $\boldsymbol{M} = \frac{2}{nm} \sum\limits_{i\in[m/2], j\in[n]} \boldsymbol{W}_{ij}$ and $\boldsymbol{U}^{\texttt{non-priv}}$ be the matrix with the top-k eigenvectors of $\boldsymbol{M}$ as columns. Let $\Pi^{\texttt{priv}} = \boldsymbol{U}^{\texttt{priv}}\left(\boldsymbol{U}^{\texttt{priv}}\right)^\top$ and $\Pi^* = \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top$. Notice that $\left\|\Pi^* - \Pi^{\texttt{priv}}\right\|_2 \leq \left\|\Pi^* - \Pi^{\texttt{non-priv}}\right\|_2 + \left\|\Pi^{\texttt{non-priv}} - \Pi^{\texttt{priv}}\right\|_2$. We bound the first term via Lemma A.1 below. In order to bound the second term, first notice that the k-th eigenvalue of $\boldsymbol{M}$ (in Algorithm 3) (denoted by $\widehat{\lambda}_k$) is lower bounded as follows. This follows with high probability from (18) by choosing appropriate β in Lemma A.1, polynomial in n^{-1}.

$$\widehat{\lambda}_k \geq \frac{\lambda_k}{d} - O\left(\sqrt{\frac{\mu^4 k^2 \lambda_k \log(dn)}{dnm}}\right) = \Omega\left(\frac{\lambda_k}{d}\right) \tag{4}$$

Now, we can use [16, Theorem 7] to directly bound $\left\|\Pi^{\texttt{non-priv}} - \Pi^{\texttt{priv}}\right\|_F = O\left(\frac{\Delta_{(\varepsilon,\delta)} d\sqrt{dk\log(dn)}}{n\cdot\lambda_k}\right)$, and correspondingly $\left\|\Pi^{\texttt{non-priv}} - \Pi^{\texttt{priv}}\right\|_2 = O\left(\frac{\zeta^2\Delta_{(\varepsilon,\delta)} d\sqrt{d\log(dn)}}{n\cdot\lambda_k}\right)$. Setting ζ as in the lemma statement, and observing rotation invariant property of the norms, completes the proof. □

Lemma A.1 (Non-private subspace closeness)**.** *Let $\Pi^{\texttt{non-priv}} = \boldsymbol{U}^{\texttt{non-priv}}\left(\boldsymbol{U}^{\texttt{non-priv}}\right)^\top$, and $\Pi^* = \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top$. Following the assumption in Lemma 4.6, we have the following for Algorithm 3 (Algorithm $\mathcal{A}_{\mathsf{Priv\text{-}init}}$) w.p. at least $1-\beta$ (over the randomness of data generation and the algorithm):*

$$\left\|\Pi^* - \Pi^{\texttt{non-priv}}\right\|_2 = \widetilde{O}\left(\sqrt{\frac{d\zeta^4\log(d/\beta)}{\lambda_k^2 nm}}\right).$$

Proof. By Gaussian concentration we have w.p. at least $1-\beta/2, \forall i \in [m], j \in [n]$, $|\langle \mathbf{x}_{ij}, \boldsymbol{U}^* \cdot \boldsymbol{v}_j^* \rangle| \leq \mu\sqrt{k\lambda_k} \cdot \sqrt{2\ln(4nm/\beta)}$ and $|\boldsymbol{z}_{ij}| \leq \sigma_{\mathrm{F}}\sqrt{2\ln(4nm/\beta)}$. Hence, if we set the clipping threshold for the response y_{ij} to be $\zeta = \left(\mu\sqrt{k\lambda_k} + \sigma_{\mathrm{F}}\right)\sqrt{2\ln(4nm/\beta)}$, then w.p. at least $1-\beta/2$, clipping will not have any impact on the analysis. Call this event $\mathcal{A}$. We will perform the linear-algebra analysis below without conditioning on this event, but our application of matrix Bernstein [49, Theorem 1.4] will rely on this bound.

We first note that for a Gaussian random vector $\mathbf{x}$, we have

$$\mathbb{E}\left[\frac{\mathbf{x}}{\|\mathbf{x}\|_2}\mathbf{x}^\top\right] = \mathbb{E}\left[\frac{\mathbf{x}\mathbf{x}^\top}{\mathbf{x}^\top\mathbf{x}}\|\mathbf{x}\|_2\right] = \frac{\mathbb{I}}{d} \cdot \mathbb{E}\left[\|\mathbf{x}\|_2\right] = \frac{\Gamma\left(\frac{d+1}{2}\right)}{d\sqrt{2}\Gamma\left(\frac{d}{2}\right)}\mathbb{I} \simeq \frac{1}{\sqrt{d}}\mathbb{I} \tag{5}$$

This can be seen by first noting that the magnitude of a random Gaussian vector is independent of its direction (i.e., the Gaussian measure with identity covariance is a product measure in spherical coordinates, trivial from the fact that it is spherically symmetric), then explicitly evaluating the expected normalized outer product $\frac{\mathbf{x}\mathbf{x}^\top}{\mathbf{x}\cdot\mathbf{x}}$. Term-by-term, this evaluation reduces to $\mathbb{E}\left[\frac{\mathbf{x}[i]\mathbf{x}[j]}{\sum_{i=1}^{d}\mathbf{x}[i]^2}\right]$. Symmetry implies this expectation is 0 for $i \neq j$ and $\frac{1}{d}$ for $i = j$. Finally we apply a well-known formula for the expected Euclidean norm of a Gaussian random vector [44]. We now have (6) and (7) (as a measure of bias and variance) for any $i \in [m/2], j \in [n]$. Here, $\left\|\boldsymbol{W}_{ij}\right\|_2$ is the operator norm of $\boldsymbol{W}_{ij}$.

$$\mathbb{E}\left[\boldsymbol{W}_{ij}\right] = \mathbb{E}\left[\frac{\mathbf{x}_{(2i)j}}{\left\|\mathbf{x}_{(2i)j}\right\|_2}\mathbf{x}_{(2i)j}^\top\left(\boldsymbol{U}^*\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\left(\boldsymbol{U}^*\right)^\top\right)\cdot\frac{\mathbf{x}_{(2i+1)j}}{\left\|\mathbf{x}_{(2i+1)j}\right\|_2}\mathbf{x}_{(2i+1)j}^\top\right] \simeq \frac{1}{d}\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top \tag{6}$$

$$\left\|\boldsymbol{W}_{ij}\right\|_2 \leq \zeta^2 \tag{7}$$

Therefore, by (6) we have the following. Here, $\boldsymbol{V}^* = [\boldsymbol{v}_1^*|\cdots|\boldsymbol{v}_n^*]$.

$$\boldsymbol{B} = \frac{4}{nm}\sum_{i\in[m/4],j\in[n]}\mathbb{E}\left[\boldsymbol{W}_{ij}\right] \simeq \boldsymbol{U}^*\left(\frac{1}{dn}\sum_{j=1}^{n}\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top = \frac{2}{dn}\boldsymbol{U}^*\left(\boldsymbol{V}^*\left(\boldsymbol{V}^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top \tag{8}$$

We will now bound $\left\|\frac{4}{nm}\sum_{i\in[m/4],j\in[n]}\boldsymbol{W}_{ij} - \boldsymbol{B}\right\|_2$ using Matrix Bernstein's inequality [48, Theorem 1.4]. Let $\boldsymbol{A}_{ij} = \boldsymbol{W}_{ij} - \frac{1}{d}\cdot\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top$. Clearly, $\mathbb{E}\left[\boldsymbol{A}_{ij}\right] = 0$, and $\left\|\boldsymbol{A}_{ij}\cdot 1_{\mathcal{A}}\right\|_2 \leq \zeta^2 + \frac{C^2}{d}$. Now, in the following we bound $\left\|\sum_{i\in[m/4],j\in[n]}\mathbb{E}\left[\boldsymbol{A}_{ij}^2\right]\right\|_2$. Let Π_j^* be the projector onto the eigenspace of $\boldsymbol{U}^*\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\left(\boldsymbol{U}^*\right)^\top$. We have the following in (9).

$$\begin{aligned}\sum_{i\in[m/4],j\in[n]}\mathbb{E}\left[\boldsymbol{A}_{ij}^2\right] &= \sum_{i\in[m/4],j\in[n]}\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right] - \frac{m}{4d^2}\sum_{j\in[n]}\boldsymbol{U}^*\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\left(\boldsymbol{U}^*\right)^\top\boldsymbol{U}^*\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\left(\boldsymbol{U}^*\right)^\top \\ &= \sum_{i\in[m/4],j\in[n]}\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right] - \frac{m}{4d^2}\sum_{j\in[n]}\left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4\cdot\Pi_j^* \end{aligned} \tag{9}$$

We now bound $\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right]$ the first term in (9). We have the following.

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right] &= \mathbb{E}\left[\frac{\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top}{\left\|\mathbf{x}_{(2i)j}\right\|_2}\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\frac{\mathbf{x}_{(2i+1)j}\mathbf{x}_{(2i+1)j}^\top}{\left\|\mathbf{x}_{(2i+1)j}\right\|_2}\frac{\mathbf{x}_{(2i+1)j}\mathbf{x}_{(2i+1)j}^\top}{\left\|\mathbf{x}_{(2i+1)j}\right\|_2}\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\frac{\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top}{\left\|\mathbf{x}_{(2i)j}\right\|_2}\right] \\ &= \mathbb{E}\left[\frac{1}{\left\|\mathbf{x}_{(2i)j}\right\|_2^2}\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\cdot\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\mathbf{x}_{(2i+1)j}\mathbf{x}_{(2i+1)j}^\top\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\right] \\ &= \mathbb{E}\left[\frac{1}{\left\|\mathbf{x}_{(2i)j}\right\|_2^2}\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\cdot\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\boldsymbol{U}^*\left(\boldsymbol{v}_j^*\left(\boldsymbol{v}_j^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\right]\end{aligned} \tag{10}$$

In the last equality, we have used independence to evaluate the outer product in the middle of the expression. This operation can be viewed as evaluating a chain of conditional expectations: $\mathbb{E}\left[\boldsymbol{A}\boldsymbol{B}\boldsymbol{A}\right] = \mathbb{E}\left[\mathbb{E}\left[\boldsymbol{A}\boldsymbol{B}\boldsymbol{A}|\boldsymbol{A}\right]\right] = \mathbb{E}\left[\boldsymbol{A}\cdot\mathbb{E}\left[\boldsymbol{B}|\boldsymbol{A}\right]\cdot\boldsymbol{A}\right] = \mathbb{E}\left[\boldsymbol{A}\cdot\mathbb{E}\left[\boldsymbol{B}\right]\cdot\boldsymbol{A}\right]$. Separating the norm of $\boldsymbol{U}^*\boldsymbol{v}_j^*(\boldsymbol{U}^*\boldsymbol{v}_j^*)^\top$ from projection onto its range, we see

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right] &= \mathbb{E}\left[\frac{\left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4}{\left\|\mathbf{x}_{(2i)j}\right\|_2^2}\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top \cdot \Pi_j^* \cdot \mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\right] \\ &= \mathbb{E}\left[\frac{\left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4}{\left\|\mathbf{x}_{(2i)j}\right\|_2^2}\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top \cdot (\Pi_j^*)^\top \cdot \Pi_j^* \cdot \mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top\right] \\ &= \left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4 \cdot \mathbb{E}\left[\left\|\Pi_j^*\mathbf{x}_{(2i)j}\right\|_2^2 \cdot \frac{\mathbf{x}_{(2i)j}\mathbf{x}_{(2i)j}^\top}{\left\|\mathbf{x}_{(2i)j}\right\|_2^2}\right] \end{aligned} \tag{11}$$

To estimate the expectation on the right, we let $\boldsymbol{a} = \Pi_j^*\mathbf{x}_{(2i)j}$ and $\boldsymbol{b} = (\mathbb{I} - \Pi_j^*)\mathbf{x}_{(2i)j}$, and note that $\boldsymbol{a}$ and $\boldsymbol{b}$ are independent. So we are interested in evaluating

$$\mathbb{E}\left[\|\boldsymbol{a}\|_2^2\,\frac{(\boldsymbol{a}+\boldsymbol{b})(\boldsymbol{a}+\boldsymbol{b})^\top}{\|\boldsymbol{a}\|_2^2+\|\boldsymbol{b}\|_2^2}\right] = \mathbb{E}\left[\frac{\|\boldsymbol{a}\|_2^2}{\|\boldsymbol{a}\|_2^2+\|\boldsymbol{b}\|_2^2}(\boldsymbol{a}\boldsymbol{a}^\top+\boldsymbol{b}\boldsymbol{b}^\top)\right] + \mathbb{E}\left[\frac{\|\boldsymbol{a}\|_2^2}{\|\boldsymbol{a}\|_2^2+\|\boldsymbol{b}\|_2^2}(\boldsymbol{a}\boldsymbol{b}^\top+\boldsymbol{b}\boldsymbol{a}^\top)\right] \tag{12}$$

The second expectation is 0, as can be noted by symmetry. That is, conditioning on $\boldsymbol{b}$ and $\|\boldsymbol{a}\|_2$ yields the integral of a spherically symmetric random variable. We can then bound:

$$\begin{aligned}\mathbb{E}\left[\|\boldsymbol{a}\|_2^2\,\frac{(\boldsymbol{a}+\boldsymbol{b})(\boldsymbol{a}+\boldsymbol{b})^\top}{\|\boldsymbol{a}\|_2^2+\|\boldsymbol{b}\|_2^2}\right] &\preccurlyeq \mathbb{E}\left[\frac{\|\boldsymbol{a}\|_2^2}{\|\boldsymbol{b}\|_2^2}\boldsymbol{a}\boldsymbol{a}^\top\right] + \mathbb{E}\left[\|\boldsymbol{a}\|_2^2\right]\mathbb{E}\left[\frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|_2^2}\right] \\ &= \mathbb{E}\left[\frac{1}{\|\boldsymbol{b}\|_2^2}\right]\mathbb{E}\left[\|\boldsymbol{a}\|_2^4\right]\Pi_j^* + \eta\left(\mathbb{I}-\Pi_j^*\right)\end{aligned} \tag{13}$$

for some $\eta > 0$. $\mathbb{E}\left[\frac{1}{\|\boldsymbol{b}\|_2^2}\right] = O\left(\frac{1}{d}\right)$ and $\mathbb{E}\left[\|\boldsymbol{a}\|_2^4\right] = O(1)$, so the first term is on the order of $\frac{1}{d}\cdot\Pi_j^*$. We evaluate η by cyclically permuting the trace:

$$\eta(d-1) = \mathbf{tr}\left(\eta\left(\mathbb{I}-\Pi_j^*\right)\right) = \mathbf{tr}\left(\mathbb{E}\left[\frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|_2^2}\right]\right) = \mathbb{E}\left[\mathbf{tr}\left(\frac{\boldsymbol{b}\boldsymbol{b}^\top}{\|\boldsymbol{b}\|_2^2}\right)\right] = \mathbb{E}\left[\mathbf{tr}\left(\frac{\boldsymbol{b}^\top\boldsymbol{b}}{\|\boldsymbol{b}\|_2^2}\right)\right] = 1 \tag{14}$$

so that $\eta = \frac{1}{d-1} = O\left(\frac{1}{d}\right)$.

Putting together (13) and (14) with (11), we see

$$\mathbb{E}\left[\boldsymbol{W}_{ij}^2\right] \preccurlyeq O\left(\frac{\left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4}{d}\right)\cdot\mathbb{I} \tag{15}$$

From (9) and (15) we have the following.

$$\left\|\sum_{i\in[m/2],j\in[n]}\mathbb{E}\left[\boldsymbol{A}_{ij}^2\right]\right\|_2 = O\left(\frac{m}{d}\sum_{j\in[n]}\left\|\boldsymbol{U}^*\boldsymbol{v}_j^*\right\|_2^4\right) = O\left(\frac{mn\mu^4k^2\lambda_k^2}{d}\right) \tag{16}$$

Therefore we may apply Matrix Bernstein's inequality [49, Theorem 1.4] by restricting nonzero values to the previously defined event $\mathcal{A}$ where clipping plays no role, ensuring the pointwise bound $\|\boldsymbol{A}_{ij}\cdot 1_{\mathcal{A}}\|_2 \leq \zeta^2 + \frac{\mu^2k\lambda_k}{d}$. Notice that this restriction can only strengthen the bound (16). So we have the following.

$$\mathbf{Pr}\left[\left\|\frac{4}{nm}\sum_{i\in[m/4],j\in[n]}\boldsymbol{A}_{ij}\cdot 1_{\mathcal{A}}\right\|_2 \geq \frac{4t}{nm}\right] \leq d\cdot\exp\left(-\frac{t^2/2}{O\left(\frac{nm\mu^4k^2\lambda_k^2}{d}\right)+\left(\zeta^2+\frac{C^2}{d}\right)\cdot\frac{t}{3}}\right) \leq \frac{\beta}{2} \tag{17}$$

Setting $t = \sqrt{\log(d/\beta)} \cdot \Omega\left(\max\left\{\sqrt{\frac{nm\mu^4k^2\lambda_k^2}{d}}, \left(\zeta^2 + \frac{\mu^2k\lambda_k}{d}\right)\sqrt{\log(d/\beta)}\right\}\right)$ in (17) suffices, by setting up and solving the associated quadratic. Therefore, since $\mathbb{P}\left[\mathcal{A}^c\right] \leq \frac{\beta}{2}$, w.p. at least $1-\beta$ we have:

$$\left\|\frac{4}{nm}\sum_{i\in[m/4],j\in[n]}\boldsymbol{A}_{ij}\right\|_2 \leq \sqrt{\log(d/\beta)}\cdot O\left(\max\left\{\frac{\mu^2k\lambda_k}{\sqrt{dnm}}, \frac{(\zeta^2+\mu^2k\lambda_k/d)\sqrt{\log(d/\beta)}}{nm}\right\}\right) = O\left(\sqrt{\frac{\zeta^4\cdot\log(d/\beta)}{dnm}}\right) \tag{18}$$

The last equality in (18) follows from the assumption $mn = \Omega\left(d\left(\zeta^2 + \frac{\mu^2k\lambda_k}{d}\right)^2 \cdot \log(d/\beta)/(\mu^2k\lambda_k)^2\right)$. With (18) in hand, we now use the Davis-Kahn Sin Θ-theorem [11] from matrix perturbation theory to bound $\left\|\Pi^{\texttt{non-priv}} - \Pi^*\right\|_2$. We use the following variant in Lemma A.2.

Lemma A.2 (Sin Θ-Theorem [11])**.** *Let $\boldsymbol{G}$ and $\boldsymbol{H}$ be two PSD matrices. Let $\Pi_{\boldsymbol{G}}^{(i)}$ be the projector onto the top-i eigenvectors of $\boldsymbol{G}$, and let $\mathsf{eig}^{(i)}(\boldsymbol{G})$ be the i-th largest eigenvalue of $\boldsymbol{G}$. Define these quantities correspondingly for $\boldsymbol{H}$. Then, the following is true.*

$$\left(\mathsf{eig}^{(i)}(\boldsymbol{G}) - \mathsf{eig}^{(j+1)}(\boldsymbol{G})\right)\cdot\left(\left(\mathbb{I} - \Pi_{\boldsymbol{H}}^{(j)}\right)\Pi_{\boldsymbol{G}}^{(i)}\right) \leq \|\boldsymbol{G}-\boldsymbol{H}\|_2$$

Let $\boldsymbol{G} = \frac{\iota}{dn}\boldsymbol{U}^*\left(\boldsymbol{V}^*\left(\boldsymbol{V}^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top$ for ι chosen via constants suppressed for clarity in (8) and $\boldsymbol{H} = \frac{4}{nm}\sum\limits_{i\in[m/4],j\in[n]}\boldsymbol{W}_{ij}$. Note that both $\boldsymbol{G}$ and $\boldsymbol{H}$ are PSD matrices. Furthermore, from (18) we have $\|\boldsymbol{G}-\boldsymbol{H}\|_2 = O\left(\sqrt{\frac{\zeta^4\cdot\log(d/\beta)}{dnm}}\right)$ w.p. $\geq 1-\beta$. Recall that $\Pi^{\texttt{non-priv}}$ is the projector onto the rank-k approximation of $\boldsymbol{H}$. Following the notation of Lemma A.2, and by assumption $\sqrt{nm} = \Omega\left(\sqrt{d\zeta^4\log(d/\beta)}\right)$, we have $\mathsf{eig}^{(k)}\left(\boldsymbol{G}\right) = \frac{\lambda_k}{d}$, $\mathsf{eig}^{(k)}\left(\Pi^{\texttt{non-priv}}\right) \in \left[\frac{\mathsf{eig}^{(k)}(\boldsymbol{G})}{2}, 2\cdot\mathsf{eig}^{(k)}\left(\boldsymbol{G}\right)\right]$, and $\mathsf{eig}^{(k+1)}\left(\Pi^{\texttt{non-priv}}\right) \leq \frac{\mathsf{eig}^{(k)}(\boldsymbol{G})}{2}$. Here, λ_k is the k-th eigenvalue of $\boldsymbol{U}^*\left(\frac{1}{n}\boldsymbol{V}^*\left(\boldsymbol{V}^*\right)^\top\right)\left(\boldsymbol{U}^*\right)^\top$, which equals the k-th eigenvalue of $\frac{1}{n}\boldsymbol{V}^*\left(\boldsymbol{V}^*\right)^\top$. Also, notice that the projector onto $\boldsymbol{G}$ equals Π^* as long as $\lambda_k > 0$, which is true by assumption.

Therefore, from Lemma A.2 we have the following w.p. at least $1-\beta$.

$$\left\|\left(\mathbb{I}-\Pi^*\right)\Pi^{\texttt{non-priv}}\right\|_2 = O\left(\frac{\sqrt{\frac{\zeta^4\cdot\log(d/\beta)}{dnm}}}{\mathsf{eig}^{(k)}\left(\boldsymbol{G}\right)}\right) \tag{19}$$

$$\left\|\left(\mathbb{I}-\Pi^{\texttt{non-priv}}\right)\Pi^*\right\|_2 = O\left(\frac{\sqrt{\frac{\zeta^4\cdot\log(d/\beta)}{dnm}}}{\mathsf{eig}^{(k)}\left(\boldsymbol{G}\right)}\right) \tag{20}$$

Furthermore, notice that $\left\|\Pi^* - \Pi^{\texttt{non-priv}}\right\|_2 \leq \left\|\left(\mathbb{I}-\Pi^*\right)\Pi^{\texttt{non-priv}}\right\|_2 + \left\|\left(\mathbb{I}-\Pi^{\texttt{non-priv}}\right)\Pi^*\right\|_2$. Plugging in the value of $\mathsf{eig}^{(k)}\left(\boldsymbol{G}\right)$ in (19) and (20) completes the proof. □

A.4 Proof of Theorem 4.2

Proof. Let $b = \langle\boldsymbol{a}, \boldsymbol{U}^*\boldsymbol{v}^*\rangle + \boldsymbol{w}$, where $\boldsymbol{a} \sim \mathcal{N}(0,1)^d$, $\boldsymbol{w} \sim \mathcal{N}(0,\sigma_{\mathrm{F}}^2)$, $\boldsymbol{U}^* \in \mathbb{R}^{d\times k}$ is a matrix with orthonormal columns, and $\boldsymbol{v}^* \in \mathbb{R}^k$. Consider the loss function $\mathcal{L}(\boldsymbol{U},\boldsymbol{v}) = \mathbb{E}_{\boldsymbol{a},\boldsymbol{w}}\left[(b - \langle\boldsymbol{a},\boldsymbol{U}\boldsymbol{v}\rangle)^2\right]$, where $\boldsymbol{U}\in\mathbb{R}^{d\times k}$ is a matrix with orthonormal columns and $\boldsymbol{v}\in\mathbb{R}^{k}$. We have,

$$\begin{aligned}
\mathcal{L}(\boldsymbol{U},\boldsymbol{v}) &= \mathbb{E}\left[\left(\boldsymbol{a}^\top\left(\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}\boldsymbol{v}\right) + \boldsymbol{w}\right)^2\right]\\
&= \left(\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}\boldsymbol{v}\right)^\top \mathbb{E}\left[\boldsymbol{a}\boldsymbol{a}^\top\right]\left(\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}\boldsymbol{v}\right) + \sigma_{\mathrm{F}}^2\\
&= \left\|\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}\boldsymbol{v}\right\|_2^2 + \sigma_{\mathrm{F}}^2. \qquad (21)
\end{aligned}$$

We consider $\widehat{\boldsymbol{v}} = \arg\min_{\boldsymbol{v}}\left\|\boldsymbol{y} - \boldsymbol{X}^\top\widehat{\boldsymbol{U}}\boldsymbol{v}\right\|_2^2 = \left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{y}$, where $\widehat{\boldsymbol{U}}\in\mathbb{R}^{d\times k}$ is some matrix with orthonormal columns, $\boldsymbol{X}\sim\mathcal{N}\left(0,1\right)^{d\times m}$ and $\boldsymbol{y} = \boldsymbol{X}^\top\boldsymbol{U}^*\boldsymbol{v}^* + \boldsymbol{w}$ (with $\boldsymbol{w}\sim\mathcal{N}(0,\sigma_{\mathrm{F}}^2)^m$. Notice that the inverse exists w.p. at least $1-\frac{1}{m^{10}}$ as long as $m=\Omega(k)$.

In the following, we will bound $\mathcal{L}(\widehat{\boldsymbol{U}},\widehat{\boldsymbol{v}})$. To do so, we will first bound $\left\|\boldsymbol{U}^*\boldsymbol{v}^* - \widehat{\boldsymbol{U}}\boldsymbol{v}\right\|_2^2$ in (21). Assume, $\widehat{\Pi} = \widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^\top$, $\Pi^* = \boldsymbol{U}^*\left(\boldsymbol{U}^*\right)^\top$, $\Delta = \widehat{\Pi} - \Pi^*$, and $\|\Delta\|_2\leq\Gamma$. We have,

$$\begin{aligned}
\mathbb{E}\left[\left\|\boldsymbol{U}^*\boldsymbol{v}^* - \widehat{\boldsymbol{U}}\widehat{\boldsymbol{v}}\right\|_2^2\right] &= \mathbb{E}\left[\left\|\widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{y} - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2\right]\\
&= \mathbb{E}\left[\left\|\widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}^*\boldsymbol{v}^* + \widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{w}\right\|_2^2\right]\\
&= \mathbb{E}\left[\left\|\widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2\right] + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&= \mathbb{E}\left[\left\|\widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\left(\widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^\top\cdot\boldsymbol{U}^*\boldsymbol{v}^* + (\mathbb{I}-\widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^\top)\boldsymbol{U}^*\boldsymbol{v}^*\right) - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2\right] + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&= \mathbb{E}\left[\left\|\widehat{\boldsymbol{U}}\left(\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\right)^{-1}\widehat{\boldsymbol{U}}^\top\boldsymbol{X}\boldsymbol{X}^\top\widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^\top\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2\right] + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&= \left\|\widehat{\boldsymbol{U}}\widehat{\boldsymbol{U}}^\top\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2 + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&= \left\|\left(\Pi^* + \Delta\right)\boldsymbol{U}^*\boldsymbol{v}^* - \boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2 + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&= \left\|\Delta\boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2 + \frac{k}{m}\sigma_{\mathrm{F}}^2\\
&\leq \Gamma^2\left\|\boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2 + \frac{k}{m}\sigma_{\mathrm{F}}^2 \qquad (22)
\end{aligned}$$

Therefore, by (22) and (21), we have the following.

$$\mathbb{E}\left[\mathcal{L}(\widehat{\boldsymbol{U}},\widehat{\boldsymbol{v}})\right] \leq \Gamma^2\left\|\boldsymbol{U}^*\boldsymbol{v}^*\right\|_2^2 + \left(\frac{k}{m}+1\right)\sigma_{\mathrm{F}}^2 \qquad (23)$$

Let $\Pi^{\mathtt{priv}} = \boldsymbol{U}^{\mathtt{priv}}\left(\boldsymbol{U}^{\mathtt{priv}}\right)^\top$. (23) immediately implies,

$$\mathsf{Risk}_{\mathsf{Pop}}\left(\left(\boldsymbol{U}^{\mathtt{priv}},\boldsymbol{V}^{\mathtt{priv}}\right);\left(\boldsymbol{U}^*,\boldsymbol{V}^*\right)\right) \leq \left\|\Pi^{\mathtt{priv}} - \Pi^*\right\|_2^2\cdot\mu^2 k\lambda_k + \left(\frac{k}{m}+1\right)\sigma_{\mathrm{F}}^2 \qquad (24)$$

Plugging in the bounds from Lemma 4.4 (and instantiating via Lemma 4.6) completes the proof. □

A.5 Proof of Lemma 4.4

Proof. Consider the t-th iteration of Algorithm 1. We first simplify the notation, i.e., let $\boldsymbol{U} = \boldsymbol{U}^{(t)}$ and $\boldsymbol{U}^{+} = \boldsymbol{U}^{(t+1)}$, $\boldsymbol{v}_j = \boldsymbol{v}_j^{(t)}$.

Now, the clipping parameters are set large enough so that under the data generation assumptions (Assumption 4.1), there is no "clipping". So the updates in the Algorithm 1 and Algorithm 2 reduce to:

$$
\begin{aligned}
\boldsymbol{v}_j &= \left(\frac{2}{m}\sum_{i\in[m/2]} \boldsymbol{U}^\top \mathbf{x}_{ij}\mathbf{x}_{ij}^\top \boldsymbol{U}\right)^{-1}\left(\frac{2}{m}\sum_{i\in[m/2]} y_{ij}\cdot \boldsymbol{U}^\top \mathbf{x}_{ij}\right),\\
\boldsymbol{H}^{(j)} &= \frac{2}{m}\sum_{i\in[m/2+1,m]} \mathbf{x}_{ij}\mathbf{x}_{ij}^\top,\\
\boldsymbol{r}^{(t)} &= \sum_{j\in\mathcal{S}_t}\left(\frac{2}{m}\sum_{i\in[m/2+1,m]} \mathbf{x}_{ij}\boldsymbol{z}_{ij}\right)\boldsymbol{v}_j^\top + \boldsymbol{g}^{(t)},\\
\widehat{\boldsymbol{U}} &= \widetilde{\mathcal{A}}^{-1}\left(\sum_{j\in\mathcal{S}_t} \boldsymbol{H}^{(j)}\boldsymbol{U}^*\boldsymbol{v}_j^*\boldsymbol{v}_j^\top + \boldsymbol{r}^{(t)}\right),\\
\boldsymbol{U}^+ &= \widehat{\boldsymbol{U}}\boldsymbol{R}^{-1},
\end{aligned}
\tag{25}
$$

where $\boldsymbol{U}^+$ and $\boldsymbol{R}$ are obtained by QR decomposition of $\widehat{\boldsymbol{U}}$. Also, $\boldsymbol{g}^{(t)} \sim \eta\zeta\Delta_{(\varepsilon,\delta)}\cdot\mathcal{N}(0,1)^{dk}$, and $\widetilde{\mathcal{A}}:\mathbb{R}^{d\times k}\to\mathbb{R}^{d\times k}$ is defined as:

$$
\begin{aligned}
&\widetilde{\mathcal{A}}(\boldsymbol{U}) = \mathcal{A}(\boldsymbol{U}) + \mathcal{G}(\boldsymbol{U}) \text{ with}\\
&\mathcal{A}(\boldsymbol{U}) = \frac{2}{m}\sum_{i\in[m/2+1,m]} \boldsymbol{H}^{(j)}\boldsymbol{U}\boldsymbol{v}_j\boldsymbol{v}_j^\top, \text{ and } \mathcal{G}(\boldsymbol{U}) = \sum_{ab}\langle \boldsymbol{G}_{ab},\boldsymbol{U}\rangle \mathbf{e}_a\mathbf{e}_b^\top,
\end{aligned}
$$

where $\mathbf{e}_a$ is the a-th standard canonical basis vector, and for $\overrightarrow{\boldsymbol{G}_{ab}}$ being the vectorized version of $\boldsymbol{G}_{ab}$, $\bar{\boldsymbol{G}} = [\overrightarrow{\boldsymbol{G}_{11}};\overrightarrow{\boldsymbol{G}_{12}};\ldots;\overrightarrow{\boldsymbol{G}_{ab}};\ldots\overrightarrow{\boldsymbol{G}_{dk}}] \sim \eta\zeta\Delta_{(\varepsilon,\delta)}\cdot\mathcal{N}_{\mathsf{sym}}(0,1)^{dk\times dk}$. Note that $\mathcal{A}$ and $\mathcal{G}$, and consequently $\widetilde{\mathcal{A}}$, are self-adjoint operator i.e. $\langle\widetilde{\mathcal{A}}(\boldsymbol{U}),\bar{\boldsymbol{U}}\rangle = \langle\boldsymbol{U},\widetilde{\mathcal{A}}(\bar{\boldsymbol{U}})\rangle$ for all $\boldsymbol{U},\bar{\boldsymbol{U}}$. Furthermore, let $\mathcal{W}(\boldsymbol{U}) = \boldsymbol{U}\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top$.

Note that the update for $\boldsymbol{v}_j$ is same as the update in the non-private Alternating Minimization algorithm (similar to Algorithm 1 of [45]). Now, let $\boldsymbol{Q} = (\boldsymbol{U}^*)^\top\boldsymbol{U}$, and $\Delta\in\mathbb{R}^{d\times k}$ be such that $\Delta_j = \boldsymbol{v}_j - \boldsymbol{Q}^{-1}\boldsymbol{v}_j^*$. Using Lemma A.4, we get:

$$
\begin{aligned}
&\|\boldsymbol{v}_j\|_2 \le \widetilde{O}\left(\frac{\mu^2 k}{n}\lambda_k^t\right), \qquad \lambda_k \le 2\lambda_k^t,\\
&\max_j \|\Delta_j\|_2 \le \widetilde{O}\left(\|(I-\boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\|\cdot\mu\sqrt{k\lambda_k}\right) + \sigma_{\mathrm{F}}\sqrt{\frac{k\log n}{m}},
\end{aligned}
\tag{26}
$$

where λ_i^t is the i-th eigenvalue of $\frac{1}{n}\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top$.

Now, using standard calculations, we get:

$$
\begin{aligned}
&\widehat{\boldsymbol{U}} - \boldsymbol{U}^*\boldsymbol{Q} \qquad (27)\\
=&\widetilde{\mathcal{A}}^{-1}\left(\sum_j \boldsymbol{H}^{(j)}\boldsymbol{U}^*\boldsymbol{Q}(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top + \sum_{ij}\boldsymbol{z}_{ij}\mathbf{x}_{ij}\boldsymbol{v}_j^\top + \boldsymbol{g}^{(t)} - \mathcal{G}(\boldsymbol{U}^*\boldsymbol{Q})\right)\\
=&\mathcal{W}^{-\frac{1}{2}}\left(\mathcal{W}^{\frac{1}{2}}\widetilde{\mathcal{A}}^{-1}\mathcal{W}^{\frac{1}{2}}\right)\mathcal{W}^{-\frac{1}{2}}\left(\sum_j \boldsymbol{H}^{(j)}\boldsymbol{U}^*\boldsymbol{Q}(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top + \sum_{ij}\boldsymbol{z}_{ij}\mathbf{x}_{ij}\boldsymbol{v}_j^\top + \boldsymbol{g}^{(t)} - \mathcal{G}(\boldsymbol{U}^*\boldsymbol{Q})\right)\\
=&\boldsymbol{U}^*\boldsymbol{Q}\sum_j(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top\left(\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top\right)^{-1} + \boldsymbol{F} + \widetilde{\boldsymbol{F}}, \qquad (28)
\end{aligned}
$$

where for $\mathcal{E} = \mathcal{W}^{\frac{1}{2}}\widetilde{\mathcal{A}}^{-1}\mathcal{W}^{\frac{1}{2}} - I$,

$$\begin{aligned}\boldsymbol{F} &= \mathcal{W}^{-\frac{1}{2}}\mathcal{E}\mathcal{W}^{-\frac{1}{2}}\left(\boldsymbol{U}^*\boldsymbol{Q}(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top\right)\\ &\quad + \mathcal{W}^{-\frac{1}{2}}\left(I + \mathcal{E}\right)\mathcal{W}^{-\frac{1}{2}}\left(\sum_j(\boldsymbol{H}^{(j)} - I)\boldsymbol{U}^*\boldsymbol{Q}(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top + \sum_{ij}\boldsymbol{z}_{ij}\mathbf{x}_{ij}\boldsymbol{v}_j^\top\right),\\ \widetilde{\boldsymbol{F}} &= \mathcal{W}^{-\frac{1}{2}}\left(I + \mathcal{E}\right)\mathcal{W}^{-\frac{1}{2}}\left(\boldsymbol{g}^{(t)} - \mathcal{G}(\boldsymbol{U}^*\boldsymbol{Q})\right).\end{aligned}$$

Using Lemma A.3 and the assumption on n, $\Delta_{(\varepsilon,\delta)}$, we get:

$$\|\mathcal{E}\|_F \leq \frac{1}{32}. \tag{29}$$

Furthermore, using Lemma A.6, we get w.p. $\geq 1 - 1/n^{100}$,

$$\|\boldsymbol{F}\|_F \leq \widetilde{O}\left(\mu \log n \cdot \sqrt{\frac{\kappa d k^2 T}{mn}}\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\|_F\right) + \sqrt{\frac{\mu^2 dkT\log n}{mn}} \cdot \frac{\sigma_{\text{F}}}{\sqrt{\lambda_k}}. \tag{30}$$

Finally, using Lemma A.7, we get w.p. $\geq 1 - 1/n^{100}$,

$$\left\|\widetilde{\boldsymbol{F}}\right\|_F \leq \widetilde{O}\left(\frac{(\sqrt{k}\eta^2 + \eta\zeta)\Delta_{(\varepsilon,\delta)}\sqrt{dk}}{n\lambda_k}\right). \tag{31}$$

That is, by setting $n = \widetilde{\Omega}\left(\frac{\lambda_1}{\lambda_k} \cdot \mu^2 dk\right)$ and $m = \widetilde{\Omega}\left((1 + \texttt{NSR}) \cdot k + k^2\right)$ (as per Assumption 4.1), we get:

$$\|\boldsymbol{F}\|_F \leq \frac{1}{64}, \left\|\widetilde{\boldsymbol{F}}\right\|_F \leq \frac{1}{64}.$$

Similarly, using n and m as specified in Assumption 4.1 and Lemma A.6, for $\boldsymbol{M} = \boldsymbol{U}^*\boldsymbol{Q}\sum_j(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top\left(\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top\right)^{-1}$, we get

$$\|\boldsymbol{M}\|_F \leq \frac{1}{64}.$$

Finally, due to the initialization condition, $\sigma_{min}(\boldsymbol{Q}) \geq 1/2$. Thus, using standard calculations (for example, see Lemma A.3 in [45]), we get:

$$\|\boldsymbol{R}^{-1}\| \leq 4,$$

where $\widehat{\boldsymbol{U}} = \boldsymbol{U}^+\boldsymbol{R}$.

Note that $\boldsymbol{U}^*\boldsymbol{Q}\sum_j(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top\left(\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top\right)^{-1}$ lies along $\boldsymbol{U}^*$, so does not contribute to the error $\left\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}^+\right\|_F$. Hence,

$$\begin{aligned}&\left\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}^+\right\|_F \leq \left\|\boldsymbol{F} + \widetilde{\boldsymbol{F}}\right\|_F\left\|\boldsymbol{R}^{-1}\right\|_F \leq 4\left\|\boldsymbol{F} + \widetilde{\boldsymbol{F}}\right\|_F\\ &\leq 4\widetilde{O}\left(\mu\log n \cdot \sqrt{\frac{\kappa dk^2T}{mn}}\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\|_F + \sqrt{\frac{\mu^2 dkT\log n}{mn}} \cdot \frac{\sigma_{\text{F}}}{\sqrt{\lambda_k}} + \frac{(\sqrt{k}\eta^2 + \eta\zeta)\Delta_{(\varepsilon,\delta)}\sqrt{dk}}{n\lambda_k}\right),\\ &\leq \frac{1}{4}\left\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\right\|_F + \widetilde{O}\left(\sqrt{\frac{\mu^2 dkT\log n}{mn}} \cdot \frac{\sigma_{\text{F}}}{\sqrt{\lambda_k}} + \frac{(\sqrt{k}\eta^2 + \eta\zeta)\Delta_{(\varepsilon,\delta)}\sqrt{dk}}{n\lambda_k}\right).\end{aligned} \tag{32}$$

The result now follows by applying the above bound for all t and by using: $\eta = \widetilde{O}(\mu\sqrt{\lambda_k dk})$, $\zeta = \widetilde{O}\left(\sigma_{\text{F}} + \mu\sqrt{k\lambda_k}\right)$, i.e., $\sqrt{k}\eta^2 + \eta\zeta = \lambda_k\widetilde{O}((\texttt{NSR} + \mu\sqrt{dk^2})\mu\sqrt{dk})$. □

Lemma A.3. *Consider the setting of Lemma 4.4 and the notation introduced in the proof above. Let $\mathcal{E} = \mathcal{W}^{\frac{1}{2}}\widetilde{\mathcal{A}}^{-1}\mathcal{W}^{\frac{1}{2}} - I$. Then, w.p. $\geq 1 - 1/n^{100}$: $\|\mathcal{E}\|_F \leq \frac{1}{32}$.*

Proof. Using Lemma A.5 and (26), we get: $\|\mathcal{W}^{-\frac{1}{2}}\mathcal{A}\mathcal{W}^{-\frac{1}{2}} - \mathcal{I}\|_F \leq 1/32$, where $\mathcal{I}(\boldsymbol{U}) = \boldsymbol{U}$. Furthermore, $\|\mathcal{W}^{-\frac{1}{2}}\mathcal{G}\mathcal{W}^{-\frac{1}{2}}\|_F \leq 8\sigma_{\texttt{Priv-1}}\sqrt{\frac{dk}{n\lambda_k}}$ by using the bound on λ_k^t given in (26). The result now follows by combining the above two given bounds. □

Lemma A.4 (Restatement of Lemma A.1 of [45])**.** *Consider the setting of Lemma 4.4 and the notation introduced in the proof above. Then, if $\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\| \leq \widetilde{O}(\frac{\lambda_k}{\lambda_1})$ and if $m \geq \widetilde{\Omega}\left((1 + \mathit{NSR}) \cdot k + k^2\right)$, we have w.p. $\geq 1 - 1/n^{101}$:*

$$\|\boldsymbol{v}_j\|_2 \leq \widetilde{O}\left(\frac{\mu^2 k}{n}\lambda_k^t\right), \quad \lambda_k \leq 2\lambda_k^t,$$

$$\max_j \|\Delta_j\|_2 \leq \widetilde{O}\left(\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U})\| \cdot \mu\sqrt{k\lambda_k}\right) + \sigma_F\sqrt{\frac{k\log n}{m}}.$$

Lemma A.5 (Restatement of Lemma A.7 of [45])**.** *Consider the setting of Lemma 4.4 and the notation introduced in the proof above. Let $mn \geq \widetilde{O}(\mu^2 dk^2)$, then w.p. $\geq 1 - 1/n^{100}$:*

$$\|\mathcal{E}\|_F \leq \widetilde{O}\left(\sqrt{\frac{\mu^2 dk^2}{mn}}\right).$$

Lemma A.6 (Restatement of Lemma A.2 of [45])**.** *Consider the setting of Lemma 4.4 and the notation introduced in the proof above. Then, if $mn \geq \widetilde{O}(\mu^2 dk^2)$, we have (w.p. $\geq 1 - 1/n^{80}$):*

$$\left\|\boldsymbol{U}^*\boldsymbol{Q}\sum_j(\boldsymbol{Q}^{-1}\boldsymbol{v}_j^* - \boldsymbol{v}_j)\boldsymbol{v}_j^\top\left(\sum_j \boldsymbol{v}_j\boldsymbol{v}_j^\top\right)^{-1}\right\|_F \leq \widetilde{O}\left(\sqrt{\kappa}\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\|_F + \frac{\sigma_F}{\sqrt{\lambda_k}} \cdot \sqrt{\frac{k}{m}}\right),$$

$$\|\boldsymbol{F}\|_F \leq \widetilde{O}\left(\mu\log n \cdot \sqrt{\frac{\kappa dk^2 T}{mn}}\|(I - \boldsymbol{U}^*(\boldsymbol{U}^*)^\top)\boldsymbol{U}\|_F\right) + \sqrt{\frac{\mu^2 dkT\log n}{mn}} \cdot \frac{\sigma_F}{\sqrt{\lambda_k}}.$$

Lemma A.7. *Consider the setting of Lemma 4.4 and the notation introduced in the proof above. Let $\|\mathcal{E}\| \leq 1/2$. Then, w.p. $\geq 1 - 1/n^{100}$:*

$$\left\|\widetilde{\boldsymbol{F}}\right\|_F \leq \widetilde{O}\left(\frac{(\sqrt{k}\eta^2 + \eta\zeta)\Delta_{(\varepsilon,\delta)}\sqrt{dk}}{n\lambda_k}\right).$$

Proof. Note that,

$$\begin{aligned}\left\|\widetilde{\boldsymbol{F}}\right\|_F &\leq \|\mathcal{W}^{-\frac{1}{2}}\left(I + \mathcal{E}\right)\mathcal{W}^{-\frac{1}{2}}\| \cdot \|\boldsymbol{g}^{(t)} - \mathcal{G}(\boldsymbol{U}^*\boldsymbol{Q})\| \leq \frac{2}{n\lambda_k}(\|\boldsymbol{g}^{(t)}\| + \|\mathcal{G}(\boldsymbol{U}^*\boldsymbol{Q})\|_F) \\ &\leq \frac{2}{n\lambda_k}(\|\boldsymbol{g}^{(t)}\| + \sqrt{k}\|\boldsymbol{G}\|_2). \end{aligned} \tag{33}$$

The lemma now follows by using the fact that: $\left\|\boldsymbol{g}^{(t)}\right\|_2 \leq \widetilde{O}(\eta\zeta\sqrt{dk})$ and $\|\boldsymbol{G}\|_2 \leq \widetilde{O}(\eta^2\sqrt{dk})$ with probability $1 - 1/n^{100}$. □

B Missing Proofs from Section 5

Proof of Theorem 5.1. We are going to proof that the sampling step in Algorithm 4 guarantees ε-DP. Let $S_0(D) = \sum\limits_{j\in[n]} \frac{2}{m} \sum\limits_{i\in[m/2]} \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_0; y_{ij}\rangle\right)$, where $\boldsymbol{U}_0$ is fixed rank-k matrix with orthonormal columns in $\mathbb{R}^{d\times k}$, and $\boldsymbol{v}_0 \in \mathbb{R}^k$, $\|\boldsymbol{v}_0\|_2 \leq C$ is a fixed vector. The sampling step in Algorithm 4 is identical to the following

$$\mathbf{Pr}[\boldsymbol{U}^{\texttt{priv}} = \boldsymbol{U}] \propto \exp\left(-\frac{\varepsilon}{8L_f C\xi} \cdot (\mathsf{score}\,(\boldsymbol{U}) - S_0(D))\right). \tag{34}$$

Let $\mathcal{L}(\boldsymbol{U}; D) = \mathsf{score}\,(\boldsymbol{U}) - S_0(D)$. Consider any neighboring data sets D and D' such that user j in D is replace by user j' in D'. We now bound the sensitivity $\mathcal{L}(\boldsymbol{U}; D) - \mathcal{L}(\boldsymbol{U}; D')$. We have

$$\begin{aligned}
&\mathcal{L}(\boldsymbol{U}; D) - \mathcal{L}(\boldsymbol{U}; D') \\
=& \left[\min_{\|\boldsymbol{v}_j\|_2 \leq C} \frac{2}{m} \sum_i \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_j \rangle; y_{ij} \right) - \frac{2}{m} \sum_i \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_0 \rangle; y_{ij} \right) \right] \\
-& \left[\min_{\|\boldsymbol{v}_{j'}\|_2 \leq C} \frac{2}{m} \sum_i \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij'}; L_f\right), \boldsymbol{v}_{j'} \rangle; y_{ij'} \right) - \frac{2}{m} \sum_i \ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij'}; L_f\right), \boldsymbol{v}_0 \rangle; y_{ij'} \right) \right]
\end{aligned} \tag{35}$$

Consider the first term. Let $\boldsymbol{v}_j^*$ be the minimizer of the first term. We have

$$\begin{aligned}
&\frac{2}{m} \sum_i \left(\ell\left(\langle \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_j^* \rangle; y_{ij} \right) - \ell(\langle \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_0 \rangle; y_{ij}) \right) \\
\leq& \frac{2}{m} \sum_i \xi \left| \langle \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_j^* \rangle - \langle \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij}; L_f\right), \boldsymbol{v}_0 \rangle \right| \\
\leq& \frac{2}{m} \sum_i \xi \left(\left\| \mathsf{clip}\left(\boldsymbol{U}^\top \mathbf{x}_{ij}; L_f\right) \right\|_2 \|\boldsymbol{v}_j^*\|_2 + \left\| \mathsf{clip}\left(\boldsymbol{U}_0^\top \mathbf{x}_{ij}; L_f\right) \right\|_2 \|\boldsymbol{v}_0\|_2 \right) \\
\leq& 2\xi L_f C,
\end{aligned}$$

where the first inequality follows because ℓ is ξ-Lipschitz in the first parameter, and the last inequality follows from the bound on the norm of $\boldsymbol{v}$. Similar can be shown for the second term of (35). Therefore, the sensitivity of the score function, i.e. (35), is upper bounded by $4\xi L_f C$.

The rest of the proof follows from standard exponential mechanism argument [34]. □

Proof of Theorem 5.2. First, to bound the size of the net $\mathcal{N}^\phi$ we use classic covering number bound from [5, Lemma 3.1]. We have $\left|\mathcal{N}^\phi\right| = O\left(\left(\frac{9\sqrt{k}}{\phi} \right)^{(2d+1)\cdot k} \right)$, since $\|\cdot\|_F$ of the matrices, over which the net is built, is $\sqrt{k}$. Let $\boldsymbol{U}^* = \arg\min_{\boldsymbol{U} \in \mathcal{K}} \mathsf{score}\,(\boldsymbol{U})$.

First, we show that $\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right) - \mathsf{score}\,(\boldsymbol{U}^*)$ is small for any $\widetilde{\boldsymbol{U}} \in \mathcal{N}^\phi$. For any $\widetilde{\boldsymbol{U}}$, we have,

$$\begin{aligned}
\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right) &\leq \mathsf{score}\,(\boldsymbol{U}^*) + \xi C \sum_{j \in [n]} \frac{2}{m} \sum_{i \in [m/2]} \left\| \mathsf{clip}\left(\widetilde{\boldsymbol{U}}^\top \mathbf{x}_{ij}; L_f \right) - \mathsf{clip}\left((\boldsymbol{U}^*)^\top \mathbf{x}_{ij}; L_f \right) \right\|_2 \\
&= \mathsf{score}\,(\boldsymbol{U}^*) + \xi C \sum_{j \in [n]} \frac{2}{m} \sum_{i \in [m/2]} \left\| \left(\widetilde{\boldsymbol{U}} - \boldsymbol{U}^* \right)^\top \mathbf{x}_{ij} \right\|_2,
\end{aligned} \tag{36}$$

with probability $\geq 1 - 1/n^{10}$. The first step follows from the Lipschitzness of ℓ and $\|\boldsymbol{v}\|_2 \leq C$, and the second step follows because the choice of L_f will not introduce any effect due to clipping w.p. at least $1 - \frac{1}{n^{10}}$. We will condition the rest of the analysis on this.

Let $\boldsymbol{M} = \widetilde{\boldsymbol{U}} - \boldsymbol{U}^*$ with columns $[\boldsymbol{m}_a : a \in [k]]$. By the definition of the net, we have $\sum_{a=1}^{k} \|\boldsymbol{m}_a\|_2^2 \leq \phi^2$. Since the feature vectors are drawn i.i.d. from $\mathcal{N}(0, 1)^d$, we have $\langle \boldsymbol{m}_a, \mathbf{x}_{ij} \rangle \sim \mathcal{N}\left(0, \|\boldsymbol{m}_a\|_2^2\right)$. Therefore, by standard Gaussian concentration and union bound, we have w.p. at least $1 - \frac{1}{n^{10}}$, $\forall i \in [m/2], j \in [n], a \in [k], |\langle \boldsymbol{m}_a, \mathbf{x}_{ij} \rangle| \leq \|\boldsymbol{m}_a\|_2 \cdot \mathrm{polylog}\,(n)$. Therefore, $\left\| \boldsymbol{M}^\top \mathbf{x}_{ij} \right\|_2 \leq \phi \cdot \mathrm{polylog}\,(n)$. Substituting back to (36), we have

$$\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right) \leq \mathsf{score}\,(\boldsymbol{U}^*) + \xi C n \phi \cdot \mathrm{polylog}\,(n). \tag{37}$$

Second, we aim to show that $\boldsymbol{U}^{\mathrm{priv}}$ and $\widetilde{\boldsymbol{U}}$ are close. For any γ, we have

$$\begin{aligned}\mathbf{Pr}\left[\mathsf{score}\left(\boldsymbol{U}^{\mathrm{priv}}\right)-\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right)\geq\gamma\right]&\leq\left|\mathcal{N}^{\phi}\right|\cdot\frac{\exp\left(-\frac{\varepsilon}{8\xi L_fC}\cdot\left(\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right)+\gamma\right)\right)}{\exp\left(-\frac{\varepsilon}{8\xi L_fC}\cdot\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right)\right)}\\&=\left|\mathcal{N}^{\phi}\right|\cdot\exp\left(-\frac{\varepsilon\gamma}{8\xi L_fC}\right).\end{aligned}\tag{38}$$

Setting γ appropriately, we have w.p. at least $1-\beta$,

$$\mathsf{score}\left(\boldsymbol{U}^{\mathrm{priv}}\right)-\mathsf{score}\left(\widetilde{\boldsymbol{U}}\right)\leq\frac{8\xi CL_f\log\left(\left|\mathcal{N}^{\phi}\right|/\beta\right)}{\varepsilon}=O\left(\frac{\xi CL_fdk}{\varepsilon}\log\left(\frac{k}{\phi\beta}\right)\right).\tag{39}$$

Now we show a bound on the excess empirical risk. Combining (37) and (39), we have

$$\mathsf{score}\left(\boldsymbol{U}^{\mathrm{priv}}\right)\leq\mathsf{score}\left(\boldsymbol{U}^{*}\right)+O\left(\frac{\xi CL_fdk}{\varepsilon}\log\left(\frac{k}{\phi\beta}\right)+\xi Cn\phi\cdot\mathrm{polylog}\left(n\right)\right).$$

Let $\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U},\boldsymbol{V})=\frac{2}{mn}\sum\limits_{i\in[m/2],j\in[n]}\ell\left(\langle\boldsymbol{U}^{\top}\mathbf{x}_{ij},\boldsymbol{v}_j\rangle;y_{ij}\right)$, and $\widehat{\boldsymbol{V}}=\min\limits_{\boldsymbol{V}}\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V})$, i.e., the minimizer for $\mathsf{score}\left(\boldsymbol{U}^{\mathrm{priv}}\right)$. The above inequality directly transfers to

$$\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U}^{\mathrm{priv}},\widehat{\boldsymbol{V}})\leq\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U}^{*},\boldsymbol{V}^{*})+O\left(\frac{\xi CL_f\cdot dk}{\varepsilon n}\log\left(\frac{k}{\phi\beta}\right)+\xi C\phi\cdot\mathrm{polylog}\left(n\right)\right)\tag{40}$$

Setting $\phi=\frac{1}{\varepsilon n}$ and plugging in $L_f=O(\sqrt{d}\log(nm))$, the above inequality becomes,

$$\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U}^{\mathrm{priv}},\widehat{\boldsymbol{V}})\leq\mathcal{L}_{\mathrm{ERM}}(\boldsymbol{U}^{*},\boldsymbol{V}^{*})+O\left(\frac{\xi C\sqrt{k^2d^3}}{\varepsilon n}\right)\cdot\mathrm{polylog}\left(n\right).\tag{41}$$

Finally, to complete the proof, we need to translate the excess empirical risk bound into excess population risk bound. Recall the following definition of population risk.

$$\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U};\boldsymbol{V})=\mathbb{E}_{(i,j)\sim_u[m/2]\times[n],(\mathbf{x}_{ij},y_{ij})\sim\tau}\left[\ell\left(\langle\boldsymbol{U}^{\top}\mathbf{x}_{ij},\boldsymbol{v}_j\rangle;y_{ij}\right)\right]\tag{42}$$

We have the following.

$$\begin{aligned}&\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}};\boldsymbol{V}^{\mathrm{priv}})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{*},\boldsymbol{V}^{*})\\&=\left(\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}};\boldsymbol{V}^{\mathrm{priv}})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{*})\right)+\left(\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{*})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{*},\boldsymbol{V}^{*})\right)\end{aligned}\tag{43}$$

We will bound the two terms separately. For the first term $\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{\mathrm{priv}})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{*})$, notice that $\boldsymbol{U}^{\mathrm{priv}}$ and $\boldsymbol{V}^{\mathrm{priv}}$ are independent as they are trained on disjoint data. This implies $\forall i\in\{m/2+1,\cdots,m\},j\in[n]$, w.p. at least $1-\frac{1}{\min\{d,n\}^{10}}$, $\left\|\left(\boldsymbol{U}^{\mathrm{priv}}\right)^{\top}\mathbf{x}_{ij}\right\|_2\leq\sqrt{k}\cdot\mathrm{polylog}\left(d,n\right)$. Since the loss functions have the form $\ell(\langle\left(\boldsymbol{U}^{\mathrm{priv}}\right)^{\top}\mathbf{x},\boldsymbol{v}\rangle;y)$, by standard uniform convergence bound [2], we have the following.

$$\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{\mathrm{priv}})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{*})=O\left(\xi C\sqrt{\frac{k}{m}}\right)\cdot\mathrm{polylog}\left(d,n\right)\tag{44}$$

Then we bound the second term $\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{\mathrm{priv}},\boldsymbol{V}^{*})-\mathcal{L}_{\mathrm{Pop}}(\boldsymbol{U}^{*},\boldsymbol{V}^{*})$ in (43). We can write the inner product $\langle\boldsymbol{U}^{\top}\mathbf{x},\boldsymbol{v}\rangle$ as $\langle\boldsymbol{U},\mathbf{x}\boldsymbol{v}^{\top}\rangle$. Therefore, if we vectorize $\boldsymbol{U}$ by concatenating its the columns as $\overrightarrow{\boldsymbol{U}}$, and vectorize $\mathbf{x}\boldsymbol{v}^{\top}$ by concatenating its columns as $\overrightarrow{\boldsymbol{z}}$, the inner product equals to $\langle\boldsymbol{z},\overrightarrow{\boldsymbol{U}}\rangle$. The loss function can be written as $\ell(\langle\boldsymbol{U}^{\top}\mathbf{x},\boldsymbol{v}\rangle;y)=\ell\left(\langle\boldsymbol{z},\overrightarrow{\boldsymbol{U}}\rangle;y\right)$. We define $\boldsymbol{z}_{ij}$ as the vectorized version of $\mathbf{x}_{ij}(\boldsymbol{v}_j^{*})^{\top}$. With probability at least $1-\frac{1}{\min\{d,n\}^{10}}$, $\forall i\in[m/2],j\in[n],\left\|\boldsymbol{z}_{ij}\right\|_2\leq$

$C\sqrt{d} \cdot \mathrm{polylog}\,(d, n)$. By standard uniform convergence bound [2] and the bound on the empirical Rademacher complexity below, we have

$$\begin{aligned}
&\mathcal{L}_{\texttt{Pop}}(\boldsymbol{U}^{\texttt{priv}}, \boldsymbol{V}^*) - \mathcal{L}_{\texttt{Pop}}(\boldsymbol{U}^*, \boldsymbol{V}^*) \\
&\leq \mathcal{L}_{\texttt{ERM}}(\boldsymbol{U}^{\texttt{priv}}, \widehat{\boldsymbol{V}}) - \mathcal{L}_{\texttt{ERM}}(\boldsymbol{U}^*, \boldsymbol{V}^*) + O\left(\xi C \sqrt{\frac{d}{nm}}\right) \cdot \mathrm{polylog}\,(d, n)\,. \qquad (45)
\end{aligned}$$

Combining (41), (45), (44) into (43) and translating the high-probability to expectation statement completes the proof.

Bound on Rademacher complexity: We aim to compute the Rademacher complexity of $\langle \boldsymbol{U}, \sum_{ij} \mathbf{x}_{ij} \boldsymbol{v}_j^\top \rangle = \sum_{ij} \langle \mathbf{x}_{ij}, \boldsymbol{U}\boldsymbol{v}_j \rangle$. We will follow [32, Theorem 11] with small modification in the Cauchy-Schwartz step.

Let θ be a vector of length nd that is formed by concatenating $\boldsymbol{U}\boldsymbol{v}_j$ for all j. For any i, j, let $\widetilde{\mathbf{x}}_{ij}$ be a vector of length dn, such that the j-th "block" (of length d) is $\mathbf{x}_{ij}$ and the rest of the entries are 0. So we can express $\langle \mathbf{x}_{ij}, \boldsymbol{U}\boldsymbol{v}_j \rangle$ as $\langle \widetilde{\mathbf{x}}_{ij}, \theta \rangle$. We have

$$\langle \widetilde{\mathbf{x}}_{ij}, \theta \rangle = \langle \mathbf{x}_{ij}, \boldsymbol{U}\boldsymbol{v}_j \rangle \leq \left\| \mathbf{x}_{ij} \right\|_2 \left\| \boldsymbol{U}\boldsymbol{v}_j \right\|_2 \leq C \left\| \mathbf{x}_{ij} \right\|_2 ,$$

where the last step follows because $\boldsymbol{U}$ is orthonormal and $\left\| \boldsymbol{v}_j \right\|_2 \leq C$. Also, because the data is drawn from a normal distribution, we have $\mathbb{E}\left[\left\| \widetilde{\mathbf{x}}_{ij} \right\|_2^2\right] = \mathbb{E}\left[\left\| \mathbf{x}_{ij} \right\|_2^2\right] = d$. The Rademacher complexity is $\frac{C\sqrt{d}}{\sqrt{mn}}$ following the same argument as [32, Theorem 11]. □