Private Multi-Task Learning: Formulation and Applications to Federated Learning

Shengyuan Hu
CMU
shengyua@andrew.cmu.edu

Zhiwei Steven Wu
CMU
zstevenwu@cmu.edu

Virginia Smith
CMU
smithv@cmu.edu

Abstract

Many problems in machine learning rely on *multi-task learning (MTL)*, in which the goal is to solve multiple related machine learning tasks simultaneously. MTL is particularly relevant for privacy-sensitive applications in areas such as healthcare, finance, and IoT computing, where sensitive data from multiple, varied sources are shared for the purpose of learning. In this work, we formalize notions of task-level privacy for MTL via *joint differential privacy* (JDP), a relaxation of differential privacy for mechanism design and distributed optimization. We then propose an algorithm for mean-regularized MTL, an objective commonly used for applications in personalized federated learning, subject to JDP. We analyze our objective and solver, providing certifiable guarantees on both privacy and utility. Empirically, we find that our method allows for improved privacy/utility trade-offs relative to global baselines across common federated learning benchmarks.

1 Introduction

Multi-task learning (MTL) aims to solve multiple learning tasks simultaneously while exploiting similarities/differences across tasks [5]. Multi-task learning is commonly used in applications that warrant strong privacy guarantees. For example, MTL has been explored in healthcare applications, as a way to learn over diverse populations or between multiple institutions [3, 21, 39]; in financial forecasting, to combine knowledge from multiple indicators or across organizations [6, 17]; and in IoT computing, as an approach for learning in federated networks of heterogeneous devices [9, 16, 19, 20, 32, 37, 38]. While MTL can significantly improve accuracy when learning in these applications, there is a dearth of work studying the privacy implications of multi-task learning.

In this work, we develop and theoretically analyze methods for MTL with formal privacy guarantees. Motivated by applications in federated learning and personalization, we aim to provide *task-level* privacy, where each task corresponds to a user/device/data silo, and the goal is to protect the sensitive information that belongs to each task's data [34].

We focus on incorporating *differential privacy* (DP) [12], which (informally) requires an algorithm's output to be insensitive to the change of any single entity's data. For MTL, using task-level DP directly would require the entire set of predictive models across all tasks to be insensitive to changes in the private data of any single task. Such a requirement is too stringent for most applications, as it implies that the predictive model for task k must have little dependence on the training data for task k, thus preventing the usefulness of the model (see Figure 1).

To circumvent this limitation, we leverage a meaningful relaxation of DP known as *joint differential privacy* (JDP) [26], which requires that for each task k, the set of output predictive models for all other tasks *except* k is insensitive to k's private data. As a consequence, it implies that the client's private data in task k is protected even if all other clients/tasks collude and share their private data and output models (as long as

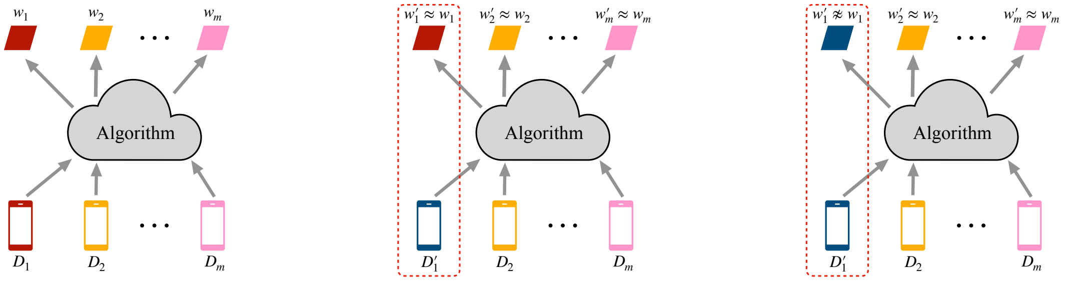


Figure 1: An MTL problem consists of m different tasks and a learning algorithm that jointly produces one model for each task (Left). For example, in cross-device federated learning, each 'task' may represent data from a mobile phone client (as depicted), and MTL can be used to learn shared, yet personalized models for each client. In traditional differential privacy, if the private data of task k (e.g., Task 1) changes, the models produced by the MTL algorithm should be indistinguishable from the models derived without changing data from task k (Middle). In contrast, JDP allows the model of task k to be dependent on task k's data while still protecting other tasks from leaking information about their private data (Right).

client k keeps their data private). In contrast to standard DP, JDP allows the predictive model for task k to depend on k's private data, helping to preserve the task's utility.

Using JDP, we then develop new learning algorithms for MTL with rigorous privacy and utility guarantees. Specifically, we propose Private Mean-Regularized MTL, a simple framework to learn multiple tasks while ensuring task-level privacy. We show that our method achieves (ϵ, δ)-JDP. Our scalable solver builds on FedAvg [33], a popular method for communication-efficient federated optimization. We analyze the convergence of the solver on both nonconvex and convex objectives, demonstrating a tradeoff between privacy and utility, and evaluate this trade-off empirically on multiple federated learning benchmarks. We summarize our contributions below:

- We provide definitions of task-level differential privacy for multi-task learning objectives (Section 3). Our definitions rely on joint differential privacy and are applicable to commonly-used multi-task relationship learning objectives.
- Using our privacy definitions, we propose Private Mean-Regularized MTL, a simple framework that provides task-level differential privacy (Section 4). We prove that our method achieves (ϵ, δ)-JDP, and we analyze the convergence of our communication-efficient solver on convex and nonconvex objectives. Our convergence analysis extends to non-private settings with partial participation, which may be of independent interest for problems in cross-device federated learning.
- Finally, we explore the performance of our approach on common federated learning benchmarks (Section 5). Our results show that it is possible to retain the accuracy benefits of MTL in these settings relative to global baselines while still providing meaningful privacy guarantees. Further, even in cases where the MTL objective achieves similar accuracy to the global objective, we find that privacy/utility benefits exist when employing the private MTL formulation.

2 Background and Related Work

Multi-task learning. Multi-task learning considers jointly solving multiple related ML tasks. Our work focuses on the general and widely-used formulation of multi-task relationship learning [44], as discussed in Section 3. This form of MTL is particularly useful in privacy-sensitive applications where datasets are shared among multiple heterogeneous entities [3, 17, 38]. In these cases, it is natural to view each data source (e.g., financial institution, hospital, mobile phone) as a separate 'task' that is learned in unison with the other tasks. This allows data to be shared, but the models to be personalized to each data silo. For example, in the setting of cross-device federated learning (described below), MTL is commonly used to train a personalized model for each device in a distributed network [30, 38].

Federated learning. A motivation for our work is the application of federated learning (FL), in which the goal is to collaboratively learn from data that has been generated by, and resides on, a number of private data silos, such as remote devices or servers [24, 29, 33]. To ensure device-level differential privacy in FL, a common technique is to learn a *global model* across the distributed data and then add noise to the aggregated model to sufficiently mask any specific client's update [15, 24, 34]. However, a defining characteristic of federated learning is that the distributed data are likely to be heterogeneous, i.e., each client may generate data via a distinct data distribution [24, 29]. To model the (possibly) varying data distributions on each client, it is natural to instead consider learning a separate model for each client's local dataset.

To this end, a number of recent works have explored multi-task learning as a way to improve the accuracy of learning in federated networks [9, 16, 19, 20, 32, 37, 38]. Despite the prevalence of multi-task federated learning, we are unaware of any work that has explored task-level privacy for commonly-used multi-task relationship models (Section 3) in federated settings.

Differentially private MTL. Prior work in private MTL differs from our own either in terms of the privacy formulation or MTL objective. For example, Wu et al. [40] explore a specific MTL setting where a feature representation shared by all tasks is first learned, followed by task-specific models on top of this private representation. We instead study multi-task relationship learning (Section 3), which is a general and widely-used MTL framework, particularly in federated learning [38].

While our work focuses on task-level privacy, there has been work on data-level privacy for MTL, which aims to protect any single piece of local data rather than protecting the entire local dataset. For example, Xie et al. [41] propose a method for data-level privacy by representing the model for each task as a sum of a public, shared weight and a task-specific weight that is only updated locally, and Gupta et al. [18] study data-level privacy for a mean estimation MTL problem.

Finally, Li et al. [28] studies multiple notions of differential privacy for meta-learning. Although similarly motivated by personalization, their framework does not cover the multi-task setting, where there exists a separate model for each task.

3 Multi-Task Learning and Privacy Formulation

In this section, we first formalize our multi-task learning objective, which is a form of mean-regularized multi-task learning (Section 3.1), and then provide our privacy formulation (Section 3.2).

3.1 Problem Setup

In the classical setting of multi-task relationship learning [43, 44], there are m different task learners with their own task-specific data. The aim is to solve:

$$\min_{W,\Omega} \left\{ F(W,\Omega) = \left\{ \frac{1}{m} \sum_{k=1}^{m} \sum_{i=1}^{n_k} l_k(x_i, w_k) + \mathcal{R}(W,\Omega) \right\} \right\} , \tag{1}$$

where w_k is the parametrization for the model of task learner k; $W = [w_1; \cdots ; w_m]$; and $\Omega \in \mathbb{R}^{m \times m}$ characterizes the relationship between every pair of task learners. A common choice for setting the regularization term $\mathcal{R}(W,\Omega)$ in previous works [38, 44] is:

$$\mathcal{R}(W,\Omega) = \lambda_1 \mathrm{tr}(W \Omega W^T) ,$$

where Ω can be viewed as a covariance matrix, used to learn/encode positive, negative, or unrelated task relationships [44]. In this paper, we focus on studying the mean-regularized multi-task learning objective [13]:

a special case of (1) where $\Omega = (\mathbf{I_{m\times m}} - \frac{1}{\mathbf{m}}\mathbf{1_m 1_m^T})^{\mathbf{2}}$ is fixed. Here $\mathbf{I_{m\times m}}$ is the identity matrix of size $m \times m$ and $\mathbf{1_m} \in \mathbb{R}^{\mathbf{m}}$ is the vector with all entries equal to 1. By picking $\lambda_1 = \frac{\lambda}{2}$, we can rewrite the objective as:

$$\min_W \left\{ F(W) = \left\{ \frac{1}{m} \sum_{k=1}^{m} \frac{\lambda}{2} \|w_k - \bar{w}\|^2 + \sum_{i=1}^{n_k} l_k(x_i, w_k) \right\} \right\}, \tag{2}$$

where $\bar{w}$ is the average of task-specific models: $\bar{w} = \frac{1}{m}\sum_{i=1}^{m} w_k$. Note that $\bar{w}$ is shared across all tasks, and each w_k is kept locally for task learner k. During optimization, each task learner k solves:

$$\min_{w_k} \left\{ f_k(w_k; \bar{w}) = \frac{\lambda}{2} \|w_k - \bar{w}\|^2 + \sum_{i=1}^{n_k} l_k(x_i, w_k) \right\}. \tag{3}$$

Despite the prevalence of this simple form of multi-task learning and its recent use in applications such as federated learning with strong privacy motivations [e.g., 10, 19, 20], we are unaware of prior work that has formalized task-level differential privacy in the context of solving Objective (2).

3.2 Privacy Formulation

We start by introducing the definition of *differential privacy (DP)* before discussing its generalization to *joint differential privacy (JDP)*. In the context of multi-task learning, each of the m task learners owns a private dataset $D_i \in \mathcal{U}_i \subset \mathcal{U}$. We define $D = \{D_1, \cdots, D_m\}$ and $D' = \{D'_1, \cdots, D'_m\}$. We call two sets D, D' *neighboring sets* if they only differ on the index i, i.e., $D_j = D'_j$ for all j except i. With this setup in mind, we define differential privacy more formally below.

Definition 1 (Differential Privacy (DP) for MTL [12]). *A randomized algorithm $\mathcal{M} : \mathcal{U}^m \to \mathcal{R}^m$ is (ϵ, δ)-differentially private if for every pair of neighboring sets that only differ in arbitrary index i: $D, D' \in \mathcal{U}$ and for every set of subsets of outputs $S \subset \mathcal{R}$,*

$$Pr(\mathcal{M}(D) \in S) \leq e^{\epsilon} Pr(\mathcal{M}(D') \in S) + \delta. \tag{4}$$

In the context of MTL, a learning algorithm then outputs one model for every task learner. As mentioned previously, since the output of MTL is a collection of models, DP would require that all the models produced by an MTL learning algorithm are insensitive to changes that happen to the private dataset of *any* single task. In this work we are interested in studying task-level privacy, where the purpose is to protect one task learner's data from leakage to any other task learners. In this setting, DP incurs an additional restriction that the model of any task learner should also be insensitive to changes in *its own data*, which would render each of the models useless. To overcome this limitation of DP, we suggest employing joint differential privacy (JDP) [26], a relaxed notion of DP, to formalize the guarantee that an MTL algorithm should provide in order to protect task-level privacy. Intuitively, JDP requires that for each task k, the set of output predictive models for all other tasks **except** k is insensitive to k's private data. We provide a formal definition below.

Definition 2 (Joint Differential Privacy (JDP) [26]). *A randomized algorithm $\mathcal{M} : \mathcal{U}^m \to \mathcal{R}^m$ is (ϵ, δ)-joint differentially private if for every i, for every pair of neighboring datasets that only differ in index i: $D, D' \in \mathcal{U}^m$ and for every set of subsets of outputs $S \subset \mathcal{R}^m$,*

$$Pr(\mathcal{M}(D)_{-i} \in S) \leq e^{\epsilon} Pr(\mathcal{M}(D')_{-i} \in S) + \delta, \tag{5}$$

where $\mathcal{M}(D)_{-i}$ represents the vector $\mathcal{M}(D)$ with the i-th entry removed.

JDP allows the predictive model for task k to depend on the private data of k, while still providing a strong guarantee: even if all the clients from all the other tasks collude and share their information, they still will not be able to learn much about the private data in the task k. JDP has mostly been used in applications related to mechanism design [8, 22, 23, 25, 26, 36]. Although it is a natural choice for achieving task-level privacy in MTL, we are unaware of any work that studies MTL subject to JDP.

We also note that we can naturally connect joint differential privacy to standard differential privacy. Informally, if we take the output of a differentially private process and run some algorithm on top of that locally for each task learner *without* communicating to the global learner or other task learners, this whole process can be shown to be joint differentially private. This is formalized as the *Billboard Lemma* [26], presented in Lemma 1 below.

Lemma 1 (Billboard Lemma)**.** *Suppose $\mathcal{M} : \mathcal{D} \to \mathcal{W}$ is (ϵ, δ)-differentially private. Consider any set of functions: $f_i : \mathcal{D}_k \times \mathcal{W} \to \mathcal{W}'$. The composition $\{f_i(\Pi_i \mathcal{D}, \mathcal{M}(\mathcal{D}))\}$ is (ϵ, δ)-joint differentially private, where $\Pi_i : \mathcal{D} \to \mathcal{D}_i$ is the projection of $\mathcal{D}$ onto $\mathcal{D}_i$.*

With the *Billboard Lemma*, we are able to obtain joint differential privacy by first training a differentially private model with data from all tasks, and then finetuning on each task with its local data. We formally introduce our algorithm and corresponding JDP guarantee by using Lemma 1 in Section 4.

Note that our privacy formulation is not limited to the multi-task relationship learning framework. For any form of multi-task learning where each task-specific model is obtained by training a combination of global component and local component, we can provide a JDP guarantee for the MTL training process by using a differentially private global component.

4 PMTL: Private Multi-Task Learning

We now present PMTL, a method for performing joint differentially-private MTL (Section 4.1). We provide both a privacy guarantee (Section 4.2) and utility guarantee (Section 4.3) for our approach.

4.1 Algorithm

We summarize our solver for private multi-task learning in Algorithm 1. Our method is based off of FedAvg [33], a communication-efficient method widely used in federated learning. FedAvg alternates between two steps: (i) each task learner selected at one communication round solves its own local objective by running stochastic gradient descent for E iterations and sending the updated model to the global learner; (ii) the global learner aggregates the local updates and broadcasts the aggregated mean. By performing local updating in this manner, FedAvg has been shown to empirically reduce the total number of communication rounds needed for convergence in federated settings relative to baselines such as mini-batch FedSGD [33]. Our private MTL algorithm differs from FedAvg in that: (i) each task learner solves the local objective with the mean-regularization term; (ii) the global learner clips individual model updates first and then adds random Gaussian noise on the aggregated model updates.

We assume that we have access to a trusted global learner[1], i.e., it is safe for a global learner to learn about task-specific data. However, since the global model is a linear combination of all task specific models and is shared among all task learners, any single task learner may infer information about other tasks from the global model. For example, in the scenario where there are only two task learners whose models are parameterized by w_1 and w_2 respectively, task learner one could simply retrieve w_2 by subtracting $\frac{1}{2}w_1$ from $\bar{w}$ at each communication round.

There are several ways in practice to overcome this privacy risk and thus achieve (ϵ, δ)-differential privacy. In this paper, we use the Gaussian Mechanism [11] during global aggregation as a simple yet effective method, highlighted in the red part on line 8 of Algorithm 1. In this case, each task learner receives a noisy aggregated global model, making it hard for any task to leak private information to the other tasks. To apply the

[1] This is a common assumption in federated learning, where access to a trusted centralized server is often assumed [24].

Algorithm 1 PMTL: Private Mean-Regularized MTL

```
1: Input: $m$, $T$, $\lambda$, $\eta_t$, $\{w_1^0, \cdots, w_m^0\}$, $\widetilde{w}^0 = \frac{1}{m}\sum_{k=1}^m w_k^0$
2: for $t = 0, \cdots, T-1$ do
3:     Global Learner randomly selects a set of tasks $S_t$ and broadcasts the mean weight $\widetilde{w}^t$
4:     for $k \in S_t$ in parallel do
5:         Each task updates its weight $w_k$ for some $E$ iterations
               $w_k^{t+1} = w_k^t - \eta_t(\nabla_{w_k^t} l_k(w_k^t) + \lambda(w_k^t - \widetilde{w}^t))$
6:         Each task sends $g_k^{t+1} = w_k^{t+1} - w_k^t$ back to the global learner.
7:     end for
8:     Global Learner computes a noisy aggregator of the weights
           $\widetilde{w}^{t+1} = \widetilde{w}^t + \frac{1}{|S_t|}\sum_{k \in S_t} g_k^{t+1} \min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \mathcal{N}(0, \sigma^2 \mathbf{I_{d\times d}})$
9: end for
10: [Optional]
11: for $k = 1, \cdots, m$ in parallel do
12:     Each task assigns $w_k = w_k^T$ and runs local finetuning for different objectives (see Section 5).
13: end for
14: return $w_1, \cdots, w_m$ as differentially private personalized models
```

Gaussian mechanism, we need to bound the ℓ_2-sensitivity of each local model update that is communicated to lie in $\mathcal{B} = \{\Delta w | \|\Delta w\|_2 \leq \gamma\}$, as highlighted in the blue part on line 8 of Algorithm 1. Hence, at each communication round, the global learner receives the model updates from each task, and clips the model updates to $\mathcal{B}$ before aggregation. After we finish training a global model, each task learner can then optionally finetune its model without communicating with the global learner. We formalize the privacy guarantee of Algorithm 1 in Section 4.2.

4.2 Privacy Analysis

We now rigorously explore the privacy guarantee provided by Algorithm 1. In our optimization scheme, for each task k, at the end of each communication round, a shared global model is received. After that the task specific model is updated by optimizing the local objective. We formalize this local task learning process as $h_k : \mathcal{D}_k \times \mathcal{W} \to \mathcal{W}$. Here we simply assume $\mathcal{W} \subset \mathbb{R}^d$ is closed. Define the mechanism for communication round t to be

$$\mathcal{M}^t(\{D_i\}, \{h_i(\cdot)\}, \widetilde{w}^t, \sigma) = \widetilde{w}^t + \frac{1}{|S_t|}\sum_{k \in S_t} h_k(D_k, \widetilde{w}^t) + \beta^t, \tag{6}$$

where $\beta^t \sim \mathcal{N}(0, \sigma^2 \mathrm{I}_{d\times d})$. Note that $\mathcal{M}^t$ characterizes a Sampled Gaussian Mechanism given $\widetilde{w}^t$ as a fixed model rather than the output of a composition of $\mathcal{M}^j$ for $j < t$. To analyze the privacy guarantee of Algorithm 1 over T communication rounds, we define the composition of $\mathcal{M}^1$ to $\mathcal{M}^T$ recursively as $\mathcal{M}^{1:T} = \mathcal{M}^T(\{D_i\}, \{h_i(\cdot)\}, \mathcal{M}^{T-1}, \sigma)$.

Theorem 1. *Assume $|S_t| = q$ for all t and the total number of communication rounds is T. There exists constants c_1, c_2 such that for any $\epsilon < c_1 \frac{q^2}{m^2} T$, the mechanism $\mathcal{M}^{1:T}$ is (ϵ, δ)-differentially private for any $\delta > 0$ if we choose $\sigma \geq c_2 \frac{\gamma\sqrt{T\log(1/\delta)}}{\epsilon m}$. When $q = m$, $\mathcal{M}^{1:T}$ is (ϵ, δ)-differentially private if we choose $\sigma = \frac{4\gamma\sqrt{T\log(1/\delta)}}{\epsilon m}$.*

Theorem 1 provides a provable privacy guarantee on the learned global model. When all tasks participate in every communication round, i.e. $q = m$, the global aggregation step in Algorithm 1 reduce to applying

Gaussian Mechanism without sampling rather than Sampled Gaussian Mechanism on the average model updates. We provide a detailed proof of Theorem 1 in Appendix A.1.

Note that Theorem 1 doesn't rely on how task learners optimize their local objective. Hence, Theorem 1 is not limited to Algorithm 1 and could be generalized to other local objectives and other global aggregation methods that produce a single model aggregate. In the next step of the training process, each task k optimizes its local objective given the private global model. We formally define this process as $h'_k : \mathcal{D}_k \times \mathcal{W} \to \mathcal{W}$. Note that h'_k is independent of h_k and could represent the optimization process for any local objective. In order to show that Algorithm 1 satisfies joint differential privacy, we would apply the *Billboard Lemma* introduced in Section 3. In our case, the average model that is broadcast by the global learner is the output of a differentially private learning process. Task learners then individually train their task specific models on the respective private data to obtain personalized models. We now present our main theorem of the JDP guarantee provided by Algorithm 1:

Theorem 2. *There exists constants c_1, c_2, for any $0 < \epsilon < c_1 \frac{q^2}{m^2} T$ and $\delta > 0$, let $\sigma \geq \frac{c_2 \gamma \sqrt{T \log(1/\delta)}}{\epsilon m}$. Algorithm 1 that outputs $h'_k(D_k, \mathcal{M}^{1:T})$ for each task is (ϵ, δ)-joint differentially private.*

From Theorem 2, we can see that for any fixed δ, the more tasks involved in the learning process, the smaller σ we need in order to keep the privacy parameter ϵ the same. In other words, less noise is required for the global model to keep the task-specific data private. When we have infinitely many tasks ($m \to \infty$), we have $\sigma \to 0$, in which case only a negligible amount of noise is needed to add to the model aggregates to make the global model private to all tasks. We provide a detailed proof in Appendix A.1.

Remark. Note that privacy guarantee provided by our Theorem 2 is not limited to mean-regularized multi-task learning. For any form of multi-task relationship learning with fixed relationship matrix Ω, as long as we fix the ℓ_2-sensitivity of model updates and the noise scale of the Gaussian mechanism applied to the statistics being broadcasted to all task learners, the privacy guarantee induced by this aggregation step is fixed, regardless of the local objective being optimized. For example, as a natural extension of our mean-regularized MTL objective, we could consider the case where task learners are partitioned into fixed clusters and optimize the mean-regularized MTL objective within each cluster, as in [14]. In this scenario, our Theorem 2 directly applies to the algorithm run on each cluster.

4.3 Convergence Analysis

As discussed in Section 3, we are interested in the following task-specific objective:

$$f_k(w_k; \widetilde{w}) = l_k(w_k) + \frac{\lambda}{2} \|w_k - \widetilde{w}\|_2^2 \tag{7}$$

where $\widetilde{w}$ is an estimate for the average model $\overline{w}$.

Here, we analyze the convergence behavior in the setting where a set S_t of q tasks participate in the optimization process at every communication round. Further, we assume the number of local optimization steps $E = 1$. We present the following convergence result:

Theorem 3 (Convergence under nonconvex loss). *Let f_k be $(L + \lambda)$-smooth. Assume γ is sufficiently large such that $\gamma \geq \max_{k,t} \|\nabla_{w_k^t} f_k(w_k^t; \widetilde{w}^t)\|_2$. Further let $f_k^* = \min_{w, \bar{w}} f_k(w; \bar{w})$ and $p = \frac{q}{m}$. If we use a fixed learning rate $\eta_t = \eta = \frac{1}{pL + (p - \frac{1}{p})\lambda}$, Algorithm 1 satisfies:*

$$\begin{aligned} \frac{1}{mT} \sum_{t=0}^{T-1} \sum_{k=1}^{m} \|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2 \leq & \frac{4\left(L + \lambda - \frac{\lambda}{p^2}\right) \sum_{k=1}^{m} (f_k(w_k^0; \widetilde{w}^0) - f_k^*)}{mT} \\ & + \frac{\mathcal{O}\left(L + \lambda + \frac{\lambda}{p^2}\right) \sum_{t=0}^{T-1} B_{t+1}}{T} + \mathcal{O}\left(Ld\lambda + d\lambda^2 + \frac{d\lambda^2}{p^2}\right) \sigma^2. \end{aligned} \tag{8}$$

where

$$B_t = \max_k f_k(w_k^t; \widetilde{w}^t). \tag{9}$$

Let σ chosen as we set in Theorem 2. Take $T = \mathcal{O}\left(\frac{m}{\lambda d\gamma^2}\right)$, the right hand side is bounded by

$$\begin{aligned}\frac{\lambda d\gamma^2}{m^2}\sum_{t=0}^{T-1}\sum_{k=1}^{m}\|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2 \leq &\frac{4\left(L+\lambda-\frac{\lambda}{p^2}\right)\lambda d\gamma^2\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{m^2}\\ &+\mathcal{O}\left(\frac{L+\lambda+\frac{\lambda}{p^2}}{m}\right)\lambda d\gamma^2\sum_{t=0}^{T-1}B_{t+1}+\mathcal{O}\left(\frac{L+\lambda+\frac{\lambda}{p^2}}{m}\right)\frac{\log(1/\delta)}{\epsilon^2}.\end{aligned} \tag{10}$$

The upper bound in Equation 8 consists of two parts: error induced by the gradient descent algorithm and error induced by the Gaussian Mechanism. When $\sigma = 0$, Algorithm 1 recovers a non-private mean-regularized multi-task learning solver.

Corollary 4. *When $\sigma = 0$, Algorithm 1 with $(L+\lambda)$-smooth and nonconvex f_k satisfies*

$$\begin{aligned}\frac{1}{mT}\sum_{t=0}^{T-1}\sum_{k=1}^{m}\|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2 \leq &\frac{4\left(L+\lambda-\frac{\lambda}{p^2}\right)\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{mT}\\ &+\frac{\mathcal{O}\left(L+\lambda+\frac{\lambda}{p^2}\right)\sum_{t=0}^{T-1}B_{t+1}}{T}.\end{aligned} \tag{11}$$

By Theorem 2, given fixed ϵ, σ^2 grows linearly with respect to T. Hence, given the same privacy guarantee, larger noise is required if the algorithm is run for more communication rounds. Note that in Theorem 3, the upper bound consists of $\mathcal{O}(\frac{1}{m\epsilon^2})$, which means when there are more tasks, the upper bound becomes smaller while the privacy parameter remains the same. On the other hand, Theorem 3 also shows a privacy-utility tradeoff using our Algorithm 1: the upper bound grows inversely proportional to the privacy parameter ϵ. We also provide a convergence analysis of Algorithm 1 with strongly-convex losses.

Theorem 5 (Convergence under strongly-convex loss)**.** *Let f_k be $(L+\lambda)$-smooth and $(\mu+\lambda)$-strongly convex. Assume γ is sufficiently large such that $\gamma \geq \max_{k,t}\|\nabla_{w_k^t} f_k(w_k^t;\widetilde{w}^t)\|_2$. Further let $w_k^* = \arg\min_w f_k(w;\bar{w}^*)$, where $\bar{w}^* = \frac{1}{m}\sum_{k=1}^m w_k^*$ and $p = \frac{q}{m}$. If we use a fixed learning rate $\eta_t = \eta = \frac{c}{\frac{L-2}{p}+\lambda p}$ for some constant c such that $0 \leq \eta p(c-2)(\mu+\lambda) \leq 1$, Algorithm 1 satisfies:*

$$\begin{aligned}\Delta_T \leq &(1-\eta p(c-2)(\mu+\lambda))^T\left(\Delta_0 - \frac{m\lambda\left(d\sigma^2+2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B}+\frac{1}{\lambda}B\right)}{\eta p(c-2)(\mu+\lambda)}\right)\\ &+\frac{m\lambda\left(d\sigma^2+2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B}+\frac{1}{\lambda}B\right)}{\eta p(c-2)(\mu+\lambda)},\end{aligned} \tag{12}$$

where $\Delta_t = f_k(w_k^t;\widetilde{w}^t) - f_k(w_k^* - \widetilde{w}^*)$ and $B = \max_t \max_k f_k(w_k^t;\widetilde{w}^t)$.

Let σ be chosen as in Theorem 2, then there exists $T = \mathcal{O}\left(\frac{m(c^2-2c)(\mu+\lambda)}{\lambda\left(\frac{L-2}{p^2}+\lambda\right)d\gamma^2}\right)$ such that

$$\begin{aligned}\Delta_T \leq &(1-\eta p(c-2)(\mu+\lambda))^T\left(\Delta_0 - \frac{\log(1/\delta)}{\epsilon^2}+\mathcal{O}\left(\frac{mB}{\eta p(c-2)(\mu+\lambda)}\right)\right)+\frac{\log(1/\delta)}{\epsilon^2}\\ &+\mathcal{O}\left(\frac{mB}{\eta p(c-2)(\mu+\lambda)}\right).\end{aligned} \tag{13}$$

Table 1: Datasets for Empirical Study

Dataset	**Number of tasks**	**Model**	**Task Type**
FEMNIST [4, 7]	205	4-layer CNN	62-class image classification
StackOverflow [1]	400	Logistic Regression	500-class tag prediction
CelebA [4, 31]	515	4-layer CNN	Binary image classification

As with Corollary 4, we recover the bound of the non-private mean-regularized MTL solver for σ=0.

Corollary 6. *When $\sigma = 0$, Algorithm 1 with $(L + \lambda)$-smooth and $(\mu + \lambda)$-strongly convex f_k satisfies*

$$\Delta_T \leq (1 - \eta p(c-2)(\mu+\lambda))^T \left(\Delta_0 - \frac{mB}{\eta p(c-2)(\mu+\lambda)}\right) + \frac{mB}{\eta p(c-2)(\mu+\lambda)}. \tag{14}$$

5 Experiments

In this section, we empirically evaluate our private MTL solver on several federated learning benchmarks. Specifically, we demonstrate the privacy-utility trade-off of training a MTL objective compared with training a global model. We then compare results of performing local finetuning after learning a MTL objective with local finetuning after learning a global objective.

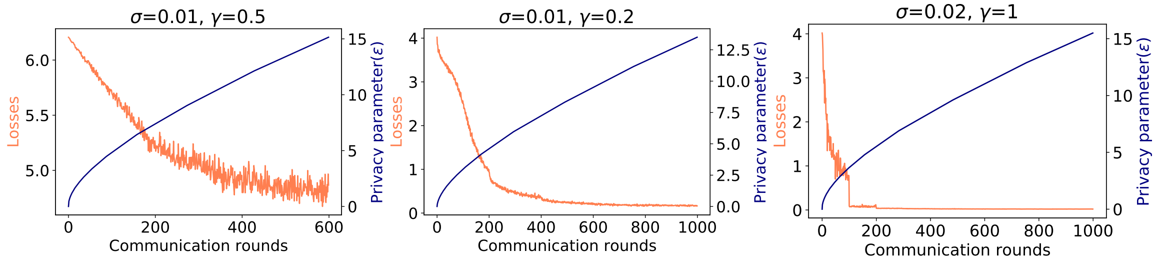
5.1 Setup

For all experiments, we evaluate the test accuracy and privacy parameter of our private MTL solver given a fixed clipping bound γ, variance of Gaussian noise σ^2, and communication rounds T. All experiments are performed on common federated learning benchmarks as a natural application of multi-task learning. We summarize the details of the datasets and models we used in our empirical study in Table 1. Our experiments include both convex (Logistic Regression) and nonconvex (CNN) loss objectives on both text (StackOverflow) and image (CelebA and FEMNIST) datasets. Each dataset is naturally partitioned among m different clients. Under such a scenario, each client can be viewed as a task and the data is only visible to the local task learner.

5.2 Privacy-Utility Trade-off of PMTL

We first explore the training loss (orange) and privacy parameter ϵ (blue) as a function of communication rounds across three datasets (Figure 2). Specifically, we evaluate the average loss for all the tasks and ϵ given a fixed δ after each round, where δ is set to be $\frac{1}{m}$ for all experiments. In general, for a fixed clipping bound γ and σ, we see that the method converges fairly quickly with respect to the resulting privacy, but that privacy guarantees may be sacrificed in order to achieve very small losses.

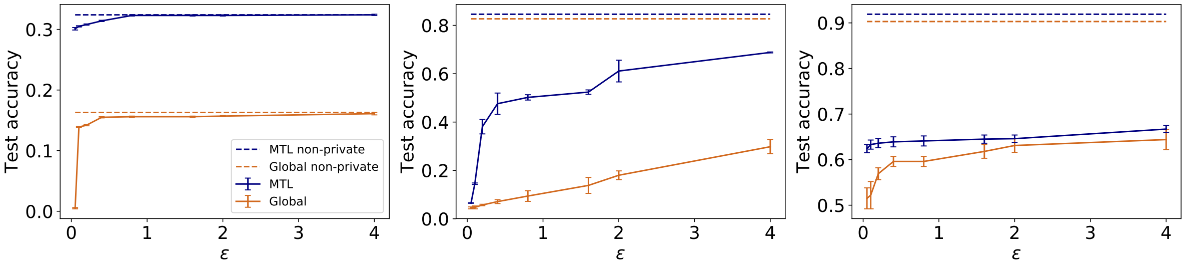
To put these results in context, we also compare the test performance of our private MTL solver with that of training a global model. In particular, we use FedAvg [33] to train a global model. At each communication round, task learners solve their local objective individually. Assuming the global learner is trustworthy, while aggregating the model updates from all tasks, the global learner applies a Gaussian Mechanism and sends the noisy aggregation back to the task learners. As a result, private FedAvg differs from our private MTL solver in the following two places: (i) the MTL objective solved locally by each task learner has an additional mean-regularized term; (ii) the MTL method evaluates on one task-specific model for every task while the global method evaluates all tasks on one global model. For each dataset, we select privacy parameter $\epsilon \in [0.05, 0.1, 0.2, 0.4, 0.8, 1.6, 2.0, 4.0]$. For each ϵ, we select the γ, σ, and T that result in the best validation



(a) StackOverflow tag prediction (b) FEMNIST (c) CelebA

Figure 2: Loss and privacy parameter vs communication rounds for PMTL. The blue line shows the change of privacy parameter ϵ in terms of number of communication rounds during training. The orange line shows the average training loss across all tasks.

accuracy for a given ϵ and record the test accuracy. A detailed description of hyperparameters is listed in Section 5.4. We plot the test accuracy with respect to the highest validation accuracy given one ϵ for both private MTL model and private global model. The results are shown in Figure 3.



(a) StackOverflow tag prediction (b) FEMNIST (c) CelebA

Figure 3: Comparison of PMTL and training a private global model.

In all three datasets, our private multi-task learning solver achieves higher test accuracy compared with training a private global model with FedAvg given the same ϵ. Moreover, the proposed mean regularized MTL solver is able to retain an advantage over global model even with noisy aggregation. In particular, for small $\epsilon < 1$, adding random Gaussian noise during global aggregation amplifies the test accuracy difference between our MTL solver and FedAvg. Under the StackOverflow tag prediction task, both methods obtain test accuracy close to the non private baseline for large ϵ.

Table 2: Comparison of private MTL and private Global model with different local finetuning methods. $\epsilon = \infty$ corresponds to the case where no noise and clipping happened, i.e., training non-privately. The higher accuracy between MTL and Global given the same ϵ and finetuning method is **bolded**.

FEMNIST	$\epsilon = 0.1$		$\epsilon = 0.8$		$\epsilon = 2.0$		$\epsilon = \infty$	
	MTL	**Global**	**MTL**	**Global**	**MTL**	**Global**	**MTL**	**Global**
Vanilla Finetuning	$\mathbf{0.645 \pm 0.013}$	0.606 ± 0.017	0.640 ± 0.016	$\mathbf{0.648 \pm 0.017}$	$\mathbf{0.677 \pm 0.008}$	0.653 ± 0.010	$\mathbf{0.832 \pm 0.005}$	0.812 ± 0.009
Mean-regularization	$\mathbf{0.608 \pm 0.011}$	0.581 ± 0.011	$\mathbf{0.605 \pm 0.008}$	0.574 ± 0.006	$\mathbf{0.656 \pm 0.009}$	0.633 ± 0.003	0.826 ± 0.011	$\mathbf{0.839 \pm 0.006}$
Symmetrized KL	$\mathbf{0.486 \pm 0.012}$	0.348 ± 0.005	$\mathbf{0.584 \pm 0.012}$	0.481 ± 0.016	$\mathbf{0.662 \pm 0.016}$	0.565 ± 0.019	$\mathbf{0.839 \pm 0.006}$	0.829 ± 0.015
EWC	$\mathbf{0.663 \pm 0.002}$	0.556 ± 0.001	0.595 ± 0.004	$\mathbf{0.607 \pm 0.007}$	$\mathbf{0.681 \pm 0.002}$	0.666 ± 0.001	$\mathbf{0.837 \pm 0.001}$	0.823 ± 0.005

5.3 PMTL with local finetuning

In federated learning, previous works have shown local finetuning with different objectives is helpful for improving utility while training a differentially private global model [42]. In this section, after obtaining a private global model, we explore locally finetuning the task specific models by optimizing different local objective functions. In particular, we use commons objectives which (i) naively optimize the local empirical risk (Vanilla Finetuning), or (ii) encourage minimizing the distance between local and global model under different distance metrics (Mean-regularization, Symmetrized KL, EWC [27, 42]). The results are listed in Table 2. When $\epsilon = \infty$ (the non-private setting), global with mean-regularization finetuning outperforms all MTL+finetuning methods. However, when we add privacy to both methods, private MTL+finetuning has an advantage over global with finetuning on different finetuning objectives. In some cases, e.g. using Symmetrized KL as the finetuning objective, the test accuracy gap between private MTL with fintuning and private global with finetuning is amplified when ϵ is small compared to the case where no privacy is added during training.

5.4 Hyperparameters

In this section we introduce how different hyperparameters affect the privacy-utility trade-off on different datasets. Each fixed privacy parameter ϵ could be computed by different combinations of noise scale σ, clipping norm γ, number of communication rounds T, and subsampling rate $p = \frac{q}{m}$. In all our experiments, we subsample 100 different tasks for each round, i.e. $q = 100$, to perform local training as well as involved in global aggregation. For FEMNIST and CelebA, we choose $\sigma \in \{0.02, 0.05, 0.1\}$ and $\gamma \in \{0.2, 0.5, 1\}$. For StackOverflow, we choose $\sigma \in \{0.01, 0.05, 0.1\}$ and $\gamma \in \{0.1, 0.5, 1\}$. We summarize both utility and privacy performance for different hyperparameters below.

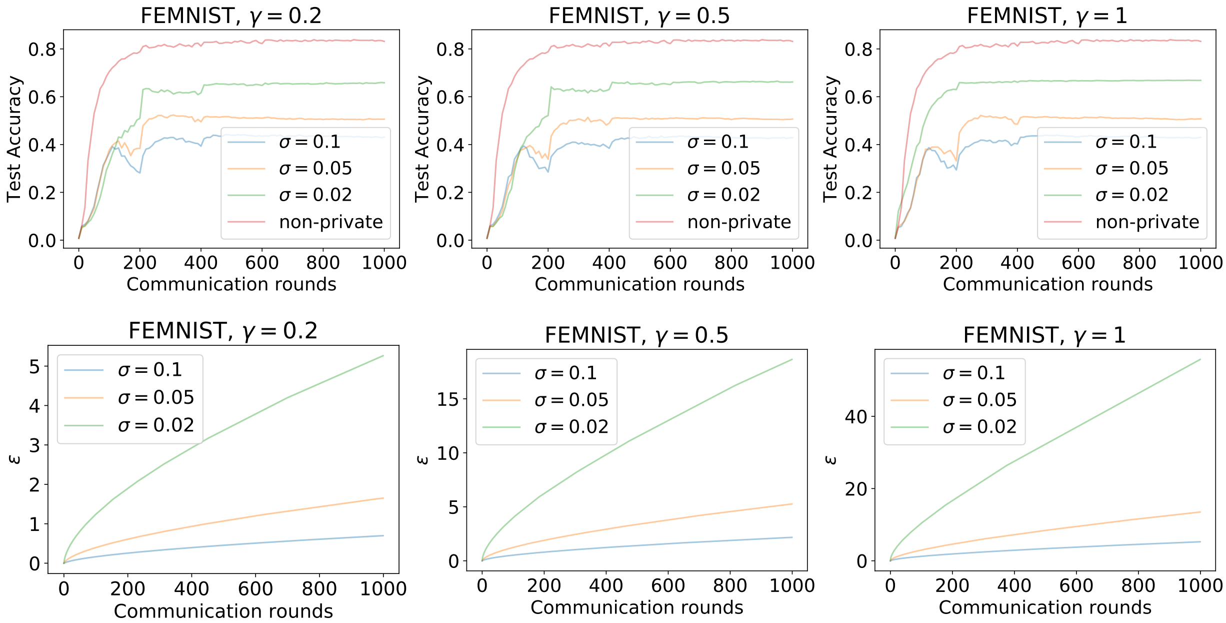


Figure 4: FEMNIST results

6 Conclusion and Future work

In this work, we define notions of task-level privacy for multi-task learning and propose a simple method for differentially private mean-regularized MTL. Theoretically, we provide both privacy and utility guarantees for our approach. Empirically, we show that private mean-regularized MTL retains advantages over training a private global on common federated learning benchmarks. In future work, we are interested in building on our results to explore privacy for more general forms of MTL, e.g., the family of objectives in (1) with arbitrary matrix Ω. We are also interested in studying how task-level privacy relates to algorithmic fairness

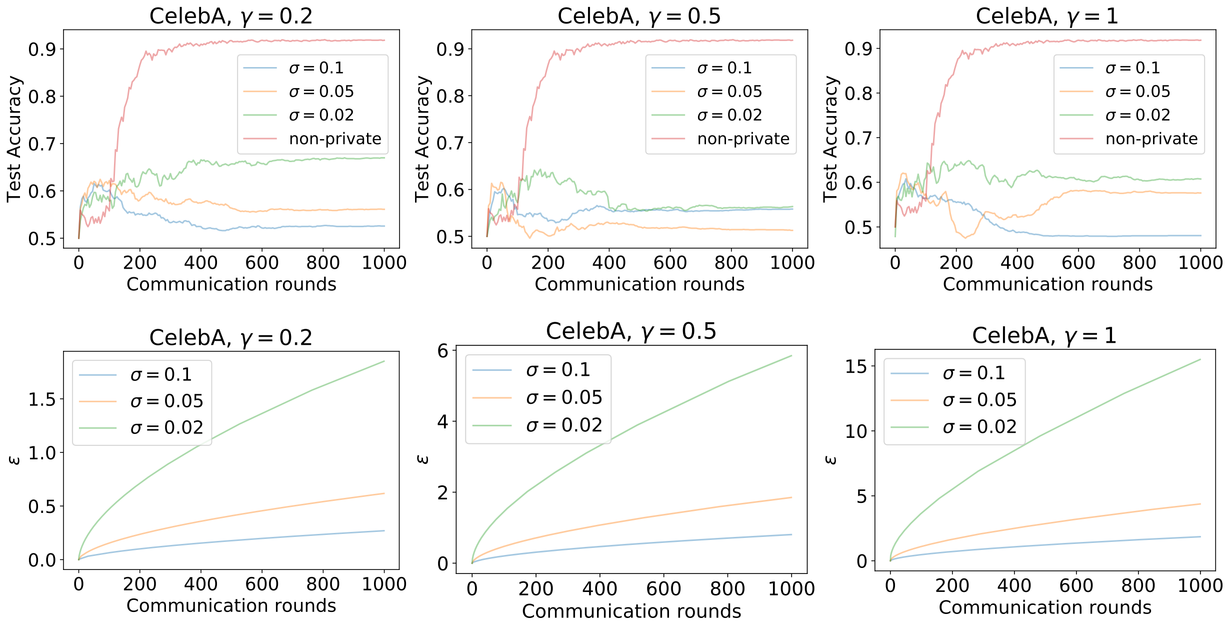


Figure 5: CelebA results

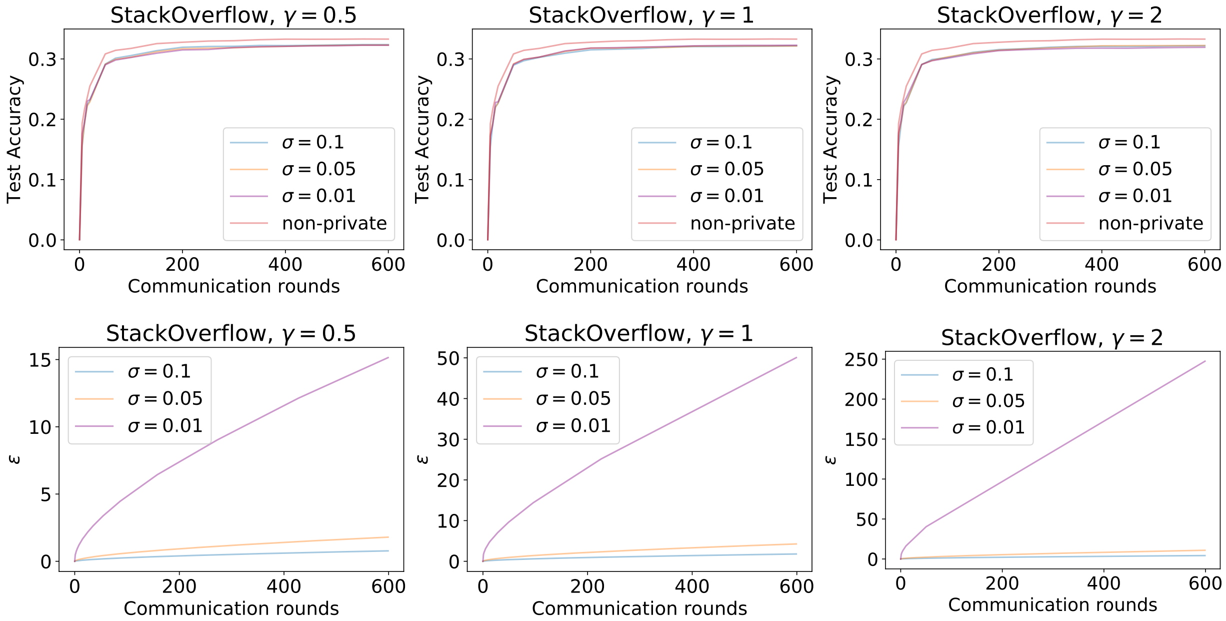


Figure 6: StackOverflow results

in the MTL setting.

7 Acknowledgements

This work was supported in part by a Google Faculty Award, Facebook Faculty Award, and Carnegie Mellon University's College of Engineering Moonshot Award for Autonomous Technologies for Livability and Sustainability (ATLAS). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect any funding agency.

References

[1] Tensorflow federated: Machine learning on decentralized data. URL `https://www.tensorflow.org/federated`.

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

[3] I. M. Baytas, M. Yan, A. K. Jain, and J. Zhou. Asynchronous multi-task learning. In *2016 IEEE 16th International Conference on Data Mining (ICDM)*, pages 11–20. IEEE, 2016.

[4] S. Caldas, P. Wu, T. Li, J. Konečnỳ, H. B. McMahan, V. Smith, and A. Talwalkar. Leaf: A benchmark for federated settings, `https://leaf.cmu.edu/`. *arXiv preprint arXiv:1812.01097*, 2018.

[5] R. Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

[6] Y. Cheng, Y. Liu, T. Chen, and Q. Yang. Federated learning for privacy-preserving ai. *Communications of the ACM*, 63(12), 2020.

[7] G. Cohen, S. Afshar, J. Tapson, and A. Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pages 2921–2926, 2017.

[8] R. Cummings, M. J. Kearns, A. Roth, and Z. S. Wu. Privacy and truthful equilibrium selection for aggregative games. In *Web and Internet Economics - 11th International Conference, WINE 2015, Amsterdam, The Netherlands, December 9-12, 2015, Proceedings*, 2015.

[9] Y. Deng, M. M. Kamani, and M. Mahdavi. Adaptive personalized federated learning. *arXiv preprint arXiv:2003.13461*, 2020.

[10] C. T. Dinh, N. H. Tran, and T. D. Nguyen. Personalized federated learning with moreau envelopes. In *Advances in Neural Information Processing Systems*, 2020.

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

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

[13] T. Evgeniou and M. Pontil. Regularized multi-task learning. In *Conference on Knowledge Discovery and Data Mining*, 2004.

[14] T. Evgeniou, C. A. Micchelli, M. Pontil, and J. Shawe-Taylor. Learning multiple tasks with kernel methods. *Journal of machine learning research*, 6(4), 2005.

[15] R. C. Geyer, T. Klein, and M. Nabi. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

[16] A. Ghosh, J. Chung, D. Yin, and K. Ramchandran. An efficient framework for clustered federated learning. *arXiv preprint arXiv:2006.04088*, 2020.

[17] J. Ghosn and Y. Bengio. Multi-task learning for stock selection. In *Advances in neural information processing systems*, pages 946–952, 1997.

[18] S. K. Gupta, S. Rana, and S. Venkatesh. Differentially private multi-task learning. In *Pacific-Asia Workshop on Intelligence and Security Informatics*, pages 101–113. Springer, 2016.

[19] F. Hanzely and P. Richtárik. Federated learning of a mixture of global and local models. *arXiv preprint arXiv:2002.05516*, 2020.

[20] F. Hanzely, S. Hanzely, S. Horváth, and P. Richtarik. Lower bounds and optimal algorithms for personalized federated learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[21] H. Harutyunyan, H. Khachatrian, D. C. Kale, G. Ver Steeg, and A. Galstyan. Multitask learning and benchmarking with clinical time series data. *Scientific data*, 6(1):1–18, 2019.

[22] J. Hsu, Z. Huang, A. Roth, T. Roughgarden, and Z. S. Wu. Private matchings and allocations. *SIAM J. Comput.*, 45(6), 2016.

[23] J. Hsu, Z. Huang, A. Roth, and Z. S. Wu. Jointly private convex programming. In *Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2016, Arlington, VA, USA, January 10-12, 2016*, 2016.

[24] P. Kairouz, H. B. McMahan, B. Avent, A. Bellet, M. Bennis, A. N. Bhagoji, K. Bonawitz, Z. Charles, G. Cormode, R. Cummings, et al. Advances and open problems in federated learning. *arXiv preprint arXiv:1912.04977*, 2019.

[25] S. Kannan, J. Morgenstern, A. Roth, and Z. S. Wu. Approximately stable, school optimal, and student-truthful many-to-one matchings (via differential privacy). In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2015, San Diego, CA, USA, January 4-6, 2015*, 2015.

[26] M. J. Kearns, M. M. Pai, A. Roth, and J. R. Ullman. Mechanism design in large games: incentives and privacy. In M. Naor, editor, *Innovations in Theoretical Computer Science, ITCS'14, Princeton, NJ, USA, January 12-14, 2014*, pages 403–410. ACM, 2014.

[27] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, 114(13):3521–3526, 2017.

[28] J. Li, M. Khodak, S. Caldas, and A. Talwalkar. Differentially private meta-learning. In *International Conference on Learning Representations*, 2019.

[29] T. Li, A. K. Sahu, A. Talwalkar, and V. Smith. Federated learning: Challenges, methods, and future directions. *IEEE Signal Processing Magazine*, 37(3):50–60, 2020.

[30] S. Liu, S. J. Pan, and Q. Ho. Distributed multi-task relationship learning. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 937–946, 2017.

[31] Z. Liu, P. Luo, X. Wang, and X. Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015.

[32] Y. Mansour, M. Mohri, J. Ro, and A. T. Suresh. Three approaches for personalization with applications to federated learning. *arXiv preprint arXiv:2002.10619*, 2020.

[33] B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial Intelligence and Statistics*, pages 1273–1282. PMLR, 2017.

[34] H. B. McMahan, D. Ramage, K. Talwar, and L. Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018.

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

[36] R. M. Rogers and A. Roth. Asymptotically truthful equilibrium selection in large congestion games. In M. Babaioff, V. Conitzer, and D. A. Easley, editors, *ACM Conference on Economics and Computation, EC '14, Stanford , CA, USA, June 8-12, 2014*, 2014.

[37] F. Sattler, K.-R. Müller, and W. Samek. Clustered federated learning: Model-agnostic distributed multitask optimization under privacy constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[38] V. Smith, C. Chiang, M. Sanjabi, and A. Talwalkar. Federated multi-task learning. In *NeurIPS*, 2017.

[39] H. Suresh, J. J. Gong, and J. V. Guttag. Learning tasks for multitask learning: Heterogenous patient populations in the icu. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 802–810, 2018.

[40] H. Wu, C. Chen, and L. Wang. A Theoretical Perspective on Differentially Private Federated Multi-task Learning. *arXiv e-prints*, art. arXiv:2011.07179, Nov. 2020.

[41] L. Xie, I. M. Baytas, K. Lin, and J. Zhou. Privacy-preserving distributed multi-task learning with asynchronous updates. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1195–1204, 2017.

[42] T. Yu, E. Bagdasaryan, and V. Shmatikov. Salvaging federated learning by local adaptation. *arXiv preprint arXiv:2002.04758*, 2020.

[43] Y. Zhang and Q. Yang. A survey on multi-task learning. *arXiv preprint arXiv:1707.08114*, 2017.

[44] Y. Zhang and D.-Y. Yeung. A convex formulation for learning task relationships in multi-task learning. In *Conference on Uncertainty in Artificial Intelligence*, 2010.

A Appendix

A.1 Privacy Analysis: Proof for Theorem 1 and 2

In the proofs of Theorem 1 and 2 we follow the line of reasoning in [2], which analyzes the privacy of DPSGD. We first state the following lemma from [2].

Lemma 2. *[2, Theorem 1] There exists constants c_1 and c_2 such that given the sampling probability $p = \frac{q}{m}$ and the number of steps T, for any $\epsilon < c_1 p^2 T$, DPSGD is (ϵ, δ)-differentially private for any $\delta > 0$ if we choose $\sigma \geq c_2 \frac{p\sqrt{T\log(1/\delta)}}{\epsilon}$.*

To prove Theorem 1, we also need the following definitions and lemmas.

Definition 3 (ℓ_2-sensitivity)**.** *Let $f : \mathcal{U} \to \mathbb{R}^d$ be some arbitrary function, the ℓ_2-sensitivity of f is defined as*

$$\Delta_2 f = \max_{adjacent\ D, D' \in \mathcal{U}} \|f(D) - f(D')\|_2 \tag{15}$$

Definition 4 (Rényi Divergence)**.** *[35, Definition 3] Let P, Q be two probability distribution over the same probability space, and let p, q be the respective probability density function. The Rényi Divergence with finite order $\alpha \neq 1$ is:*

$$D_\alpha(P\|Q) = \frac{1}{\alpha - 1} \ln \int_{\mathcal{X}} q(x) \left(\frac{p(x)}{q(x)}\right)^\alpha dx \tag{16}$$

Definition 5 ((α, ϵ)-Rényi Differential Privacy)**.** *[35, Definition 4] A randomized mechanism $f : \mathcal{D} \to \mathcal{R}$ is said to have (α, ϵ)-Rényi Differential Privacy if for all adjacent $D, D' \in \mathcal{D}$ it holds that:*

$$D_\alpha(f(D)\|f(D')) \leq \epsilon. \tag{17}$$

Lemma 3. *[35, Corollary 3] The Gaussian mechanism is $(\alpha, \alpha(2(\Delta_2 f)^2/\sigma^2))$-Renyi Differentially Private.*

Lemma 4. *[35, Proposition 3] If f is (α, ϵ)-RDP, then it is $(\epsilon + \frac{\log(1/\delta)}{\alpha - 1}, \delta)$-DP for all $\delta > 0$.*

We begin by proving the first part of Theorem 1, where $q \neq m$.

Proof for Theorem 1: $q \neq m$. Note that aggregation step in line 8 of Algorithm 1 can be rewritten as

$$\begin{aligned}
\widetilde{w}^{t+1} &= \widetilde{w}^t + \frac{1}{|S_t|} \sum_{k \in S_t} g_k^{t+1} \min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \mathcal{N}(0, \sigma^2 \mathbf{I_{d\times d}}) && (18) \\
&= \widetilde{w}^t + \frac{1}{q} \sum_{k \in S_t} g_k^{t+1} \min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \mathcal{N}\left(0, \left(\frac{\sigma}{\gamma}\right)^2 \gamma^2 \mathbf{I_{d\times d}}\right) && (19) \\
&= \widetilde{w}^t + \frac{1}{q} \left(\sum_{k \in S_t} g_k^{t+1} \min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \mathcal{N}\left(0, \left(\frac{q\sigma}{\gamma}\right)^2 \gamma^2 \mathbf{I_{d\times d}}\right)\right). && (20)
\end{aligned}$$

From here, we can directly apply Lemma 2 with σ set to be $\frac{q\sigma}{\gamma}$. Hence, we conclude that when $q \neq m$, there exists constants c_1 and c_2 such that given the number of steps T, for any $\epsilon < c_1 \frac{q^2}{m^2} T$, $\mathcal{M}^{1:T}$ is (ϵ, δ)-differentially private for any $\delta > 0$ if we choose $\sigma \geq c_2 \frac{\gamma\sqrt{T\log(1/\delta)}}{m\epsilon}$. □

This proof can extend to the case where $q = m$. In the remainder of this section, we provide a proof that gives a more specific bound on the variance σ^2 in the case where $q = m$.

Proof for Theorem 1: $q = m$. Define $H^t : \prod_{i=1}^m \mathcal{D}_i \times \mathcal{W} \to \mathcal{W}$ as

$$H^t(\{D_i\}, \{h_i(\cdot)\}, \widetilde{w}^t) = \widetilde{w}^t + \frac{1}{m}\sum_{i=1}^m h_i^t(D_i, \widetilde{w}^t). \tag{21}$$

As a result, we have $\mathcal{M}^t(\{D_i\}, \{h_i(\cdot)\}, \widetilde{w}^t, \sigma) = H^t(\{D_i\}, \{h_i(\cdot)\}, \widetilde{w}^t) + \beta^t$.

By Lemma 3, $\mathcal{M}^t$ is $(\alpha, 2\alpha(\Delta_2 H^t)^2/d\sigma^2)$-Renyi Differentially Private. Note that

$$\begin{aligned}
(\Delta_2 H^t)^2 &= \max_j \max_{\text{adjacent } D_j, D_j' \in \mathcal{D}_j} \left\| H^t(\{D_1, \cdots, D_j, \cdots, D_m\}) - H^t(\{D_1, \cdots, D_j', \cdots, D_m\}) \right\|^2 && (22)\\
&= \max_j \max_{\text{adjacent } D_j, D_j' \in \mathcal{D}_j} \left\| \frac{1}{m} h_j^t(D_j, \widetilde{w}^t) - \frac{1}{m} h_j^t(D_j', \widetilde{w}^t) \right\|^2 && (23)\\
&= \frac{1}{m^2} \max_j \max_{\text{adjacent } D_j, D_j' \in \mathcal{D}_j} \left\| h_j^t(D_j, \widetilde{w}^t) - h_j^t(D_j', \widetilde{w}^t) \right\|^2 && (24)\\
&= \frac{1}{m^2} \max_j (\Delta_2 h_j^t)^2. && (25)
\end{aligned}$$

Hence, by sequential composition of Rényi Differential Privacy [35, Proposition 1], $\mathcal{M}^{1:T}$ is $(\alpha, \sum_{i=1}^T 2\alpha \max_j (\Delta_2 h_j^t)^2/m^2\sigma^2)$-RDP.

By Lemma 4, we know that $\mathcal{M}^{1:T}$ is $(\sum_{i=1}^T 2\alpha \max_j (\Delta_2 h_j^t)^2/m^2\sigma^2 + \frac{\log(1/\delta)}{\alpha - 1}, \delta)$-DP.

Plugging in $\alpha = \frac{4\log(1/\delta)}{\epsilon}$, $\sigma = \frac{4\gamma\sqrt{T\log(1/\delta)}}{\epsilon m}$, we have

$$\begin{aligned}
\sum_{i=1}^T 2\alpha \max_j (\Delta_2 h_j^t)^2/m^2\sigma^2 + \frac{\log(1/\delta)}{\alpha - 1} &\le \sum_{i=1}^T 2\alpha\gamma^2/m^2\sigma^2 + \frac{\log(1/\delta)}{\alpha - 1} && (26)\\
&= \frac{2\frac{4\log(1/\delta)}{\epsilon}\gamma^2}{m^2(\frac{4\gamma\sqrt{T\log(1/\delta)}}{\epsilon m})^2} + \frac{\log(1/\delta)}{\frac{4\log(1/\delta)}{\epsilon} - 1} && (27)\\
&\le \frac{\epsilon}{2} + \frac{\epsilon}{2} && (28)\\
&= \epsilon. && (29)
\end{aligned}$$

Hence, $\mathcal{M}^{1:T}$ is (ϵ, δ)-JDP if we choose $\sigma = \frac{4\gamma\sqrt{T\log(1/\delta)}}{\epsilon m}$. □

By Theorem 1 and *Billboard Lemma*, it directly follows that Algorithm 1 is $(\epsilon, \delta)-$JDP.

Proof for Theorem 2. Theorem 1 shows that Algorithm 1 consists of a (ϵ, δ)-DP process to produce global model. After that each task learner trains local model with the DP global model and its private data. By Lemma 1, it directly follows that Algorithm 1 is (ϵ, δ)-JDP. □

A.2 Convergence Analysis(nonconvex):

Proof for Theorem 3. Let $w_k^* = \arg\min_w f_k(w; \bar{w}^*)$. Let I_k^t be the random variable indicating whether task k is selected in communication round t. Note that the probability task learner k is selected in any arbitrary communication round $p_k = \frac{\binom{m-1}{q-1}}{\binom{m}{q}} = \frac{q}{m}$. Thus $\mathbb{E}[I_k^t] = p_k = \frac{q}{m}$. By L-smoothness of f_k, we have

$$\begin{aligned}
\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^t) - f_k(w_k^t;\widetilde{w}^t)] &\leq \mathbb{E}\left[\langle \nabla f_k(w_k^t;\widetilde{w}^t), w_k^{t+1} - w_k^t\rangle + \frac{L}{2}\|w_k^{t+1} - w_k^t\|^2\right] && (30)\\
&= \mathbb{E}\left[\langle \nabla f_k(w_k^t;\widetilde{w}^t), \eta_t I_k^t \nabla f_k(w_k^t;\widetilde{w}^t)\rangle + \frac{L}{2}\|\eta_t I_k^t \nabla f_k(w_k^t;\widetilde{w}^t)\|^2\right] && (31)\\
&= \left(\frac{L+\lambda}{2}\eta_t^2 p_k^2 - \eta_t p_k\right)\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2. && (32)
\end{aligned}$$

Hence, we have

$$\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^{t+1}) - f_k(w_k^t;\widetilde{w}^t)] \leq \underbrace{\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^{t+1}) - f_k(w_k^{t+1};\widetilde{w}^t)]}_{\mathrm{B}} + \left(\frac{L}{2}\eta_t^2 p_k^2 - \eta_t p_k\right)\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2. \tag{33}$$

It suffices to bound B:

$$\begin{aligned}
\mathrm{B} &= \mathbb{E}\left[\frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 - \frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^t\|^2\right] && (34)\\
&= \frac{\lambda}{2}\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|\|2w_k^{t+1} - \widetilde{w}^t - \widetilde{w}^{t+1}\|] && (35)\\
&\leq \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]\mathbb{E}[\|2w_k^{t+1} - \widetilde{w}^t - \widetilde{w}^{t+1}\|^2]} && (36)\\
&= \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}\sqrt{\mathbb{E}[\|(\widetilde{w}^{t+1} - \widetilde{w}^t) + 2(w_k^{t+1} - \widetilde{w}^{t+1})\|^2]} && (37)\\
&\leq \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2] + 4\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 + 4\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|\|w_k^{t+1} - \widetilde{w}^{t+1}\|]} && (38)\\
&\leq \frac{\lambda}{2}\sqrt{\underbrace{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}_{\mathrm{C}_1}}\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2] + 4\underbrace{\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2}_{\mathrm{C}_2} + 4\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2]\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2}} && (39)
\end{aligned}$$

where the first and third inequality follows from Cauchy-Schwartz Inequality: $\mathbb{E}[XY] \leq \sqrt{\mathbb{E}[X^2]\mathbb{E}[Y^2]}$. We can then upper bound C_1 and C_2.

$$\mathrm{C}_1 = \mathbb{E}\left[\left\|\frac{1}{q}\sum_{k\in S_t} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \beta^t\right\|^2\right] \tag{40}$$

$$\leq \left(\sqrt{\mathbb{E}\left[\left\|\frac{1}{q}\sum_{k=1}^{m} I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{\mathbb{E}[\|\beta^t\|^2]}\right)^2 \tag{41}$$

$$= \left(\sqrt{\mathbb{E}\left[\left\|\frac{1}{q}\sum_{k=1}^{m} I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{42}$$

$$\leq \left(\sqrt{\frac{m}{q^2}\sum_{k=1}^{m}\mathbb{E}\left[\left\|I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{43}$$

$$\leq \left(\sqrt{\frac{m}{q^2}\sum_{k=1}^{m}\mathbb{E}\left[\left\|I_k^{t+1}\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{44}$$

$$\leq \left(\sqrt{\frac{1}{m}\sum_{k=1}^{m}\left\|\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2} + \sqrt{d}\sigma\right)^2. \tag{45}$$

Denote $h(t) = \sqrt{\frac{1}{m}\sum_{k=1}^{m}\left\|\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2}$. We have:

$$\mathrm{C}_2 \leq \frac{2}{\lambda}\frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 \tag{46}$$

$$\leq \frac{2}{\lambda} f_k(w_k^{t+1}; \widetilde{w}^{t+1}) \tag{47}$$

$$= \frac{2}{\lambda} B_{t+1} \tag{48}$$

Plugging the bounds for C_1 and C_2 into B yields:

$$\mathrm{B} \leq \frac{\lambda}{2}(h(t) + \sqrt{d}\sigma)\left(h(t) + \sqrt{d}\sigma + 2\sqrt{\frac{2}{\lambda}B_{t+1}}\right). \tag{49}$$

Denote the right hand side as $\beta(t)$, we have

$$\mathbb{E}[f_k(w_k^{t+1}; \widetilde{w}^{t+1}) - f_k(w_k^t; \widetilde{w}^t)] \leq \beta(t) + \left(\frac{L+\lambda}{2}\eta_t^2 p_k^2 - \eta_t p_k\right)\|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2. \tag{50}$$

Let $\delta_t = \mathbb{E}[f_k(w_k^t; \widetilde{w}^t) - f_k(w_k^*; \bar{w}^*)]$, we have

$$\delta_{t+1} \leq \delta_t + \beta(t) + \left(\frac{L+\lambda}{2}\eta_t^2 p_k^2 - \eta_t p_k\right)\|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2. \tag{51}$$

In the nonconvex case, we have

$$\sum_{t=0}^{T-1}\left(\eta_t p_k - \frac{L+\lambda}{2}\eta_t^2 p_k^2\right)\|\nabla f_k(w_k^t; \widetilde{w}^t)\|^2 - \beta(t) \leq f_k(w_k^0; \widetilde{w}^0) - f_k^* \tag{52}$$

Summing over k on the left handed side, when γ is large enough so that no clipping happens we have

$$
\begin{aligned}
&\sum_{k=1}^{m}\sum_{t=0}^{T-1}\left(\eta_t p_k - \frac{L+\lambda}{2}\eta_t^2 p_k^2\right)\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 - \beta(t) && (53)\\
&=\sum_{t=0}^{T-1}\left(\eta_t \frac{q}{m} - \frac{L+\lambda}{2}\eta_t^2 \frac{q^2}{m^2}\right)\sum_{k=1}^{m}\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 - m\beta(t) && (54)\\
&= \begin{array}{l} \displaystyle\sum_{t=0}^{T-1}\left(\eta_t \frac{q}{m} - \frac{L+\lambda}{2}\eta_t^2 \frac{q^2}{m^2}\right)\sum_{k=1}^{m}\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 \\ \displaystyle -\frac{\lambda}{2}\left(mh^2(t) + \left(2\sqrt{d}\sigma + 2\sqrt{\frac{2}{\lambda}B_{t+1}}\right)mh(t) + m\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right)\right)\end{array} && (55)\\
&=\sum_{t=0}^{T-1}\left(\eta_t \frac{q}{m} - \frac{L+\lambda}{2}\eta_t^2 \frac{q^2}{m^2} - \frac{\lambda}{2}\eta_t^2\right)G_t^2 - \lambda\sqrt{m}\left(\sqrt{d}\sigma + \sqrt{\frac{2}{\lambda}B_{t+1}}\right)\eta_t G_t - \frac{\lambda m}{2}\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right) && (56)\\
&\le \sum_{k=1}^{m} f_k(w_k^0;\widetilde{w}^0) - f_k^*, && (57)
\end{aligned}
$$

where $G_t = \sqrt{\sum_{k=1}^{m}\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2}$. Picking $\eta_t = \frac{mq}{qL-(m-q)\lambda}$ yields

$$
\begin{aligned}
&\sum_{t=0}^{T-1}\frac{q^2}{2(q^2L-(m^2-q^2)\lambda)}G_t^2 - \frac{\lambda\sqrt{m}\left(\sqrt{d}\sigma + \sqrt{\frac{2}{\lambda}B_{t+1}}\right)mq}{qL-(m-q)\lambda}G_t - \frac{\lambda m}{2}\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right) && (58)\\
&\le \sum_{k=1}^{m} f_k(w_k^0;\widetilde{w}^0) - f_k^*. && (59)
\end{aligned}
$$

This is equivalent to

$$
\begin{aligned}
&\sum_{t=0}^{T-1}G_t^2 - 2\lambda\sqrt{m}\left(\sqrt{d}\sigma + \sqrt{\frac{2}{\lambda}B_{t+1}}\right)\frac{m}{q}G_t - \left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\lambda m\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right) && (60)\\
&\le 2\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^{m} f_k(w_k^0;\widetilde{w}^0) - f_k^*. && (61)
\end{aligned}
$$

Hence, we have

$$
\begin{aligned}
&\sum_{t=0}^{T-1}\left(G_t - \lambda m\left(\sqrt{d}\sigma + \sqrt{\frac{2}{\lambda}B_{t+1}}\right)\frac{m}{q}\right)^2 && (62)\\
&\le 2\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^{m} f_k(w_k^0;\widetilde{w}^0) - f_k^* + \sum_{t=0}^{T-1}(L\lambda m + m\lambda^2)\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right) + 2\frac{m^3\lambda B_{t+1}}{q^2}. && (63)
\end{aligned}
$$

This implies

$$
\begin{aligned}
&\sum_{t=0}^{T-1}G_t^2 && (64)\\
&\le 2\left(2\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^{m} f_k(w_k^0;\widetilde{w}^0) - f_k^* + \sum_{t=0}^{T-1}(L\lambda m + m\lambda^2 + \frac{2\lambda^2 m^3}{q^2})\left(d\sigma^2 + 2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}\right) + 4\frac{m^3\lambda B_{t+1}}{q^2}\right). && (65)
\end{aligned}
$$

Hence, we conclude that

$$
\begin{aligned}
&\frac{1}{T}\sum_{t=0}^{T-1}\sum_{k=1}^{m}\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 && (66)\\
&\le \frac{4\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{T}+\frac{\mathcal{O}\left(L\lambda m+\lambda^2 m+\frac{\lambda^2 m^3}{q^2}\right)\sum_{t=0}^{T-1}\left(d\sigma^2+2\sigma\sqrt{\frac{2d}{\lambda}B_{t+1}}+\frac{2B_{t+1}}{\lambda}\right)}{T} && (67)\\
&\le \frac{4\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{T}+\frac{\mathcal{O}\left(Lm+\lambda m+\frac{\lambda m^3}{q^2}\right)\sum_{t=0}^{T-1}\left(\sqrt{d\lambda}\sigma+\sqrt{2B_{t+1}}\right)^2}{T} && (68)\\
&\le \frac{4\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{T}+\frac{\mathcal{O}\left(L+\lambda+\frac{\lambda m^2}{q^2}\right)m\sum_{t=0}^{T-1}B_{t+1}}{T}+\mathcal{O}\left(Ld\lambda+d\lambda^2+\frac{d\lambda^2m^2}{q^2}\right)m\sigma^2. && (69)
\end{aligned}
$$

Taking $\sigma=\frac{c_2\gamma\sqrt{T\log(1/\delta)}}{m\epsilon}$ and $T=\mathcal{O}\left(\frac{m}{\lambda d\gamma^2}\right)$, we have

$$
\begin{aligned}
&\frac{1}{mT}\sum_{t=0}^{T-1}\sum_{k=1}^{m}\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 && (70)\\
&\le \frac{4\left(L+\lambda-\frac{m^2}{q^2}\lambda\right)\sum_{k=1}^m(f_k(w_k^0;\widetilde{w}^0)-f_k^*)}{mT}+\frac{\mathcal{O}\left(L+\lambda+\frac{m^2}{q^2}\lambda\right)\sum_{t=0}^{T-1}B_{t+1}}{T}+\frac{1}{m}\mathcal{O}\left(L+\lambda+\frac{m^2}{q^2}\lambda\right)\frac{\log(1/\delta)}{\epsilon^2}. && (71)
\end{aligned}
$$

□

A.3 Convergence Analysis(Convex):

Proof for Theorem 5. Let $w_k^*=\arg\min_w f_k(w;\bar{w}^*)$. Let I_k^t be the random variable indicating whether task k is selected in communication round t. Thus $\mathbb{E}[I_k^t]=p_k$. By $L+\lambda$-smoothness and $\mu+\lambda$-strong convexity of f_k, we have

$$
\begin{aligned}
\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^t)-f_k(w_k^t;\widetilde{w}^t)] &\le \mathbb{E}\left[\langle\nabla f_k(w_k^t;\widetilde{w}^t),w_k^{t+1}-w_k^t\rangle+\frac{L}{2}\|w_k^{t+1}-w_k^t\|^2\right] && (72)\\
&=\mathbb{E}\left[\langle\nabla f_k(w_k^t;\widetilde{w}^t),\eta_t I_k^t\nabla f_k(w_k^t;\widetilde{w}^t)\rangle+\frac{L}{2}\|\eta_t I_k^t\nabla f_k(w_k^t;\widetilde{w}^t)\|^2\right] && (73)\\
&=\left(\frac{L+\lambda}{2}\eta_t^2p_k^2-\eta_tp_k\right)\|\nabla f_k(w_k^t;\widetilde{w}^t)\|^2 && (74)\\
&\le\left(\frac{L+\lambda}{2}\eta_t^2p_k^2-\eta_tp_k\right)2(\mu+\lambda)(f(w_k^t;\widetilde{w}^t)-f(w_k^*;\widetilde{w}^t)) && (75)\\
&\le\left(\frac{L+\lambda}{2}\eta_t^2p_k^2-\eta_tp_k\right)2(\mu+\lambda)(f(w_k^t;\widetilde{w}^t)-f(w_k^*;\widetilde{w}^*)). && (76)
\end{aligned}
$$

Hence, we have

$$
\begin{aligned}
\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^{t+1})-f_k(w_k^t;\widetilde{w}^t)]\le&\underbrace{\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^{t+1})-f_k(w_k^{t+1};\widetilde{w}^t)]}_{\mathrm{B}}\\
&+\left((L+\lambda)\eta_t^2p_k^2-2\eta_tp_k\right)(\mu+\lambda)(f(w_k^t;\widetilde{w}^t)-f_k^*).
\end{aligned} \tag{77}
$$

It suffices to bound B:

$$\mathrm{B} = \mathbb{E}\left[\frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 - \frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^t\|^2\right] \tag{78}$$

$$= \frac{\lambda}{2}\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|\|2w_k^{t+1} - \widetilde{w}^t - \widetilde{w}^{t+1}\|] \tag{79}$$

$$\leq \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]\mathbb{E}[\|2w_k^{t+1} - \widetilde{w}^t - \widetilde{w}^{t+1}\|^2]} \tag{80}$$

$$= \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}\sqrt{\mathbb{E}[\|(\widetilde{w}^{t+1} - \widetilde{w}^t) + 2(w_k^{t+1} - \widetilde{w}^{t+1})\|^2]} \tag{81}$$

$$\leq \frac{\lambda}{2}\sqrt{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2] + 4\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 + 4\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|\|w_k^{t+1} - \widetilde{w}^{t+1}\|]} \tag{82}$$

$$\leq \frac{\lambda}{2}\sqrt{\underbrace{\mathbb{E}[\|\widetilde{w}^t - \widetilde{w}^{t+1}\|^2]}_{\mathrm{C}_1}}\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2] + 4\underbrace{\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2}_{\mathrm{C}_2} + 4\sqrt{\mathbb{E}[\|\widetilde{w}^{t+1} - \widetilde{w}^t\|^2]\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2}} \tag{83}$$

where the first and third inequality follows from Cauchy-Schwartz Inequality: $\mathbb{E}[XY] \leq \sqrt{\mathbb{E}[X^2]\mathbb{E}[Y^2]}$. It suffices to find the upper bound of C_1 and C_2.

$$\mathrm{C}_1 = \mathbb{E}\left[\left\|\frac{1}{q}\sum_{k\in S_t} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right) + \beta^t\right\|^2\right] \tag{84}$$

$$\leq \left(\sqrt{\mathbb{E}\left[\left\|\frac{1}{q}\sum_{k=1}^{m} I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{\mathbb{E}[\|\beta^t\|^2]}\right)^2 \tag{85}$$

$$= \left(\sqrt{\mathbb{E}\left[\left\|\frac{1}{q}\sum_{k=1}^{m} I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{86}$$

$$\leq \left(\sqrt{\frac{m}{q^2}\sum_{k=1}^{m}\mathbb{E}\left[\left\|I_k^{t+1} g_k^{t+1}\min\left(1, \frac{\gamma}{\|g_k^{t+1}\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{87}$$

$$\leq \left(\sqrt{\frac{m}{q^2}\sum_{k=1}^{m}\mathbb{E}\left[\left\|I_k^{t+1}\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2\right]} + \sqrt{d}\sigma\right)^2 \tag{88}$$

$$\leq \left(\sqrt{\frac{1}{m}\sum_{k=1}^{m}\left\|\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2} + \sqrt{d}\sigma\right)^2. \tag{89}$$

Denote $h(t) = \sqrt{\frac{1}{m}\sum_{k=1}^{m}\left\|\eta_t\nabla f_k(w_k^t)\min\left(1, \frac{\gamma}{\eta_t\|\nabla f_k(w_k^t)\|_2}\right)\right\|^2}$. On the other hand,

$$\mathrm{C}_2 \leq \frac{2}{\lambda}\frac{\lambda}{2}\|w_k^{t+1} - \widetilde{w}^{t+1}\|^2 \tag{90}$$

$$\leq \frac{2}{\lambda}f_k(w_k^{t+1}; \widetilde{w}^{t+1}) \tag{91}$$

$$= \frac{2}{\lambda}B_{t+1}. \tag{92}$$

Plug the bounds for C_1 and C_2 into B:

$$\mathrm{B} \leq \frac{\lambda}{2}(h(t) + \sqrt{d}\sigma)\left(h(t) + \sqrt{d}\sigma + 2\sqrt{\frac{2}{\lambda}B_{t+1}}\right) \tag{93}$$

$$\leq \lambda\left(h^2(t) + d\sigma^2 + 2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B_{t+1}} + \frac{1}{\lambda}B_{t+1}\right) \tag{94}$$

Denoting the right hand side as $\beta(t)$, we have

$$\mathbb{E}[f_k(w_k^{t+1};\widetilde{w}^{t+1}) - f_k(w_k^t;\widetilde{w}^t)] \leq \beta(t) + \left((L+\lambda)\eta_t^2 p_k^2 - 2\eta_t p_k\right)(\mu+\lambda)(f(w_k^t;\widetilde{w}^t) - f_k^*). \tag{95}$$

Letting $\delta_k^t = \mathbb{E}[f_k(w_k^t;\widetilde{w}^t) - f_k(w_k^*;\bar{w}^*)]$, we have

$$\delta_k^{t+1} \leq \left(1 - \left((L+\lambda)\eta_t^2 p_k^2 - 2\eta_t p_k\right)(\mu+\lambda)\right)\delta_k^t + \beta(t). \tag{96}$$

Summing over k on the left handed side, when γ is large enough so that no clipping happens we have

$$\sum_{k=1}^m \delta_k^{t+1} \tag{97}$$

$$\leq \left(1 - \left((L+\lambda)\eta_t^2 p^2 - 2\eta_t p\right)(\mu+\lambda)\right)\sum_{k=1}^m \delta_k^t + m\beta(t) \tag{98}$$

$$= \left(1 - \left((L+\lambda p^2 - 2)\eta_t^2 - 2\eta_t p\right)(\mu+\lambda)\right)\sum_{k=1}^m \delta_k^t + m\lambda\left(d\sigma^2 + 2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B_{t+1}} + \frac{1}{\lambda}B_{t+1}\right). \tag{99}$$

Let $\Delta_t = \sum_{k=1}^m \delta_k^t$. Assume $\max_{t\leq T} B_t = B$. Pick $C = \frac{m\lambda\left(d\sigma^2 + 2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B} + \frac{1}{\lambda}B\right)}{((L+\lambda p^2-2)\eta_t^2 - 2\eta_t p)(\mu+\lambda)}$, we have

$$\Delta_{t+1} - C \leq \left(1 - \left((L+\lambda p^2 - 2)\eta_t^2 - 2\eta_t p\right)(\mu+\lambda)\right)(\Delta_t - C). \tag{100}$$

Choose $\eta_t = \eta = \frac{c}{\frac{L-2}{p}+\lambda p}$ for some constant c such that $0 < \left(1 - \left((L+\lambda p^2-2)\eta_t^2 - 2\eta_t p\right)(\mu+\lambda)\right) < 1$. Apply recursively to all t, we obtain

$$\Delta_T \leq \left(1 - \frac{(c^2-2c)(\mu+\lambda)}{\frac{L-2}{p^2}+\lambda}\right)^T\left(\Delta_0 - \frac{m\lambda\left(d\sigma^2 + 2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B} + \frac{1}{\lambda}B\right)}{\frac{(c^2-2c)(\mu+\lambda)}{\frac{L-2}{p^2}+\lambda}}\right) + \frac{m\lambda\left(d\sigma^2 + 2\sqrt{d}\sigma\sqrt{\frac{2}{\lambda}B} + \frac{1}{\lambda}B\right)}{\frac{(c^2-2c)(\mu+\lambda)}{\frac{L-2}{p^2}+\lambda}}. \tag{101}$$

Take $\sigma = \frac{c_2\gamma\sqrt{T\log(1/\delta)}}{m\epsilon}$ and we can find $T = \mathcal{O}\left(\frac{m(c^2-2c)(\mu+\lambda)}{\lambda\left(\frac{L-2}{p^2}+\lambda\right)d\gamma^2}\right)$ such that,

$$\Delta_T \leq \left(1 - \frac{(c^2-2c)(\mu+\lambda)}{\frac{L-2}{p^2}+\lambda}\right)^T\left(\Delta_0 - \frac{\log(1/\delta)}{\epsilon^2} + \mathcal{O}\left(\frac{mB\left(\frac{L-2}{p^2}+\lambda\right)}{(c^2-2c)(\mu+\lambda)}\right)\right) + \frac{\log(1/\delta)}{\epsilon^2} + \mathcal{O}\left(\frac{mB\left(\frac{L-2}{p^2}+\lambda\right)}{(c^2-2c)(\mu+\lambda)}\right). \tag{102}$$

□