A New Analysis Framework for Federated Learning on Time-Evolving Heterogeneous Data

Yongxin Guo [1] **Tao Lin** [2] **Xiaoying Tang** [1 3]

Abstract

Federated Learning (FL) is an emerging learning paradigm that preserves privacy by ensuring client data locality on edge devices. The optimization of FL is challenging in practice, due to the diversity and heterogeneity of the learning system. In spite of recent research efforts on improving the optimization on heterogeneous data, the effects of time-evolving heterogeneous data has not been well studied. In this paper, we propose a unified data heterogeneity setup by considering time-evolving heterogeneity across edge devices as the first step to capture the real-world FL scenarios. We introduce a new theoretical framework, Continual Federated Learning (CFL), to integrate different aspects of data heterogeneity in both FL and conventional Continual Learning (CL). Towards understanding the proposed novel, realistic, yet difficult FL scenario, we design a regularization-based algorithm CFL-R. We show that CFL-R has a better convergence rate than Federated Averaging (FedAvg). Empirically, CFL-R significantly outperforms other FL baselines and can alleviate the issue of catastrophic forgetting.

1. Introduction

Federated Learning (FL) is a distributed machine learning method that preserves privacy by ensuring client data locality on edge devices. As the workhorse algorithm in FL, FedAvg performs multiple local stochastic gradient descent (SGD) updates on the available clients before communicating with the server (McMahan et al., 2017). However, FedAvg suffers from the large heterogeneity (non-IID) in the data presented on the different clients, which leads to slow and unstable convergence (Karimireddy et al., 2020b). To tackle this problem, a line of research has been proposed, which either simulates the distribution of the entire dataset with preassigned weights of clients (Wang et al., 2020; Reisizadeh et al., 2020; Mohri et al., 2019; Li et al., 2020b) or adopts variance reduction techniques (Karimireddy et al., 2020b;a; Das et al., 2020; Haddadpour et al., 2021). However, all these methods assume a fixed data distribution across clients over all training rounds, ignoring the fact that this assumption does not always hold in practice: both random selection and users' interaction can cause time-evolving data heterogeneity.

Continual Learning (CL) is a closely related learning paradigm on a similar time-evolving scenario: it aims to learn from sequence data that are time-varying and incrementally available (French, 1999; Kirkpatrick et al., 2017). However, most of the prior CL works do not consider the (distributed) collaborative learning, and the theoretical understanding of these CL works is quite limited.

In this paper, we consider the time-evolving heterogeneity across edge devices and propose a new framework to capture the formulations of both classical FL and CL. We propose a new regularization-based algorithm CFL-R, and show its fast convergence (than FedAvg) theoretically and empirically. Besides, we are the first to prove the convergence rate of mini-batch SGD in CL scenario, whereas the existing work, e.g., Yin et al. (2020b), only considers the full batch gradient descent.

Contribution. As our key contributions, we

- introduce a new analysis framework, Continual Federated Learning (CFL), to formulate time-evolving heterogeneous data in FL;
- propose a new regularization based algorithm CFL-R and show it empowers a better convergence rate than FedAvg;
- give a novel convergence rate for mini-batch SGD in CL;
- show the fast convergence and robustness (to the catastrophic forgetting) of CFL-R over other FL competitors, through extensive empirical experiments.

2. Related Work

The theoretical analysis on the convergence of FedAvg can date back to the parallel SGD analysis on the identical functions (Zinkevich et al., 2010) and recently is refined by (Stich, 2019; Stich & Karimireddy, 2020; Patel &

[1]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen [2]EPFL [3]The Shenzhen Institute of Artificial Intelligence and Robotics for Society. Correspondence to: Xiaoying Tang <tangxiaoying@cuhk.edu.cn>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

Dieuleveut, 2019; Khaled et al., 2020; Woodworth et al., 2020b). For the analysis on heterogeneous data, Li et al. (2020c) first give the convergence rate of FedAvg on non-iid datasets with random selection, and assume that the client optimum is ϵ-close. Woodworth et al. (2020a); Khaled et al. (2020) give tighter convergence rates under the assumption of bounded gradient drift.More recently, Karimireddy et al. (2020b); Koloskova et al. (2020); Yang et al. (2021) give the convergence analysis of local SGD for non-convex objective functions under bounded gradient noise assumptions.

For Continual Learning, there exists a large number of works. We concentrate on regularization based methods (Kirkpatrick et al., 2017; Li & Hoiem, 2017; Zenke et al., 2017). However, the efforts on the theoretical understanding are limited. A very recent preprint (Yin et al., 2020b) provides a viewpoint of regularization-based continual learning by formulating it as a second-order Taylor approximation of the loss function of each task, which can derive many existing algorithms.

To the best of our knowledge, the scenario of CFL first appeared in Bui et al. (2018), and FedWeIT (Yoon et al., 2020) extends the regularization-based method to enable learning a sequence of client tasks; however, their approach has no theoretical guarantee. A more in-depth discussion of related work is given in Appendix D.

3. Continual Federated Learning Framework

3.1. Scenarios with Time-Evolving Heterogeneous Data

In this section, we introduce two scenarios that will lead to time-evolving heterogeneity.

1. **Scenario 1 (S1: The difference between the clients selected per round).** In some federated learning scenarios, thousands of clients will participate in the entire training process, while we choose only a very small portion of them in each round of training. Therefore, most of these clients may participate in training only once. In this case, for two consecutive rounds of training, the clients we selected are likely to be unrelated, or even have completely opposite preferences.

2. **Scenario 2 (S2: The local datasets of the same clients evolve).** Even for the same clients, when users delete old data or generate new data, the distribution of their local datasets can also be different. For example, users can generate different amounts of data in different times. Besides, the proportion of the labels in the local datasets may be different at different times.

The above-mentioned two scenarios lead to time-evolving data heterogeneity, but with distinctions. In scenario S1, the data heterogeneity comes from the different chosen clients in different training rounds, while in scenario S2, the reason for data distribution migration is that the client data often changes over time. Note that both scenarios may co-exist within real-world cases.

Due to the time-evolving data heterogeneity, the assumption on the fixed data distribution (over time) in most of the existing work cannot precisely describe the above scenarios. Thus, it is necessary to formulate a new optimization problem that can capture the time-evolving data heterogeneity that emerged in the training process.

3.2. Formulation

Conventional Federated Learning. The conventional FL normally minimize a sum-structured distributed optimization problem $f : \mathbb{R}^d \to \mathbb{R}$, of the form

$$f^\star := \min_{\boldsymbol{\omega}\in\mathbb{R}^d} \left[f(\boldsymbol{\omega}) := \sum_{i=1}^N p_i f_i(\boldsymbol{\omega}) \right], \tag{1}$$

where the components $f_i(\boldsymbol{\omega}) : \mathbb{R}^d \to \mathbb{R}$ are distributed among N nodes/clients and are given in a stochastic form: $f_i(\boldsymbol{\omega}) := \mathbb{E}_{\mathcal{D}_i}[F_i(\boldsymbol{\omega})]$ on node $i \in [N]$. In each communication round of FedAvg, the node/client i receives the model parameters from the server, and performs K local SGD update steps in the form of $\boldsymbol{\omega}_{t,i,k} = \boldsymbol{\omega}_{t,i,k-1} - \eta_l(\nabla f_i(\boldsymbol{\omega}_{t,i,k-1}) + \boldsymbol{\nu}_{t,i,k-1})$ with local step-size η_l and gradient noise $\boldsymbol{\nu}_{t,i,k-1}$. Then the client communicates its update $\Delta\boldsymbol{\omega}_{t,i} = \boldsymbol{\omega}_{t,i,K} - \boldsymbol{\omega}_t$ with the server for the model aggregation: $\boldsymbol{\omega}_{t+1} = \boldsymbol{\omega}_t - \frac{\eta_g}{N}\sum_{i=1}^N \Delta\boldsymbol{\omega}_{t,i}$.

Continual Federated Learning. Notice that the above FL formulation cannot precisely capture the fact that local client data often changes over time in real-world scenario. Considering the time-evolving heterogeneous data, we formulate the true objective function of CFL as below.

$$f(\boldsymbol{\omega}) = \mathbb{E}[f_{t,i}(\boldsymbol{\omega})] = \sum_{t=1}^{T}\sum_{i=1}^{N} p_{t,i} f_{t,i}(\boldsymbol{\omega}), \tag{2}$$

where $f_{t,i}(\boldsymbol{\omega})$ is the local objective function of client i on round t. Note that the CFL formulation integrates the generic forms of both FL and CL formulations. Setting $T = 1$ reduces to the generic form of FL formulation, due to the fixed local datasets (across time), while $N = 1$ recovers the generic form of CL formulation for a centralized model.

In practical, in round t, we can't get access to local datasets of round τ for any $\tau > t$. Thus, the objective functions we indeed optimize in round t is,

$$\bar{f}_{t,i}(\boldsymbol{\omega}) = \sum_{\tau=1}^{t} p_{\tau,i} f_{\tau,i}(\boldsymbol{\omega}), \tag{3}$$

$$\bar{f}_t(\boldsymbol{\omega}) = \sum_{i=1}^{N} \bar{f}_{t,i}(\boldsymbol{\omega}). \tag{4}$$

Here $\bar{f}_{t,i}(\boldsymbol{\omega})$ denotes the local objective function of round t, while $\bar{f}_t(\boldsymbol{\omega})$ denotes the global objective function.

In scenario S1, clients will only appear once in training procedure. In scenario S2, users have rights to delete old data. Thus, we can't always get access to local datasets of

previous rounds. To tackle this difficulty, inspired by the regularization based techniques (Yin et al., 2020a) in CL, we use the second order form of Taylor series expansion to approximate the local objective function of round t. Then to approximate $\bar{f}_{t,i}(\boldsymbol{\omega})$, we introduce $\tilde{f}_{t,i}(\boldsymbol{\omega})$.

$$\begin{aligned}\tilde{f}_{t,i}(\boldsymbol{\omega}) &= p_{t,i} f_{t,i}(\boldsymbol{\omega}) \\ &+ \sum_{\tau=1}^{t-1} p_{\tau,i} \left(f_{\tau,i}(\hat{\boldsymbol{\omega}}_{\tau,i}) + \nabla f_{\tau,i}(\hat{\boldsymbol{\omega}}_{\tau,i})^T (\boldsymbol{\omega} - \hat{\boldsymbol{\omega}}_{\tau,i}) \right) \\ &+ \sum_{\tau=1}^{t-1} p_{\tau,i} \left(\frac{1}{2} (\boldsymbol{\omega} - \hat{\boldsymbol{\omega}}_{\tau,i})^T \hat{\mathbf{H}}_{\tau,i} (\boldsymbol{\omega} - \hat{\boldsymbol{\omega}}_{\tau,i}) \right) , \quad (5)\end{aligned}$$

$$\tilde{f}_t(\boldsymbol{\omega}) = \sum_{i=1}^{N} f_{t,i}(\boldsymbol{\omega}) , \quad (6)$$

where $\hat{\boldsymbol{\omega}}_{\tau,i}$ denotes the parameters of client i at the end of round τ, and $\hat{\boldsymbol{\omega}}_{\tau,i} = \omega_{\tau,i,K}$ if we have K local epochs each round. Similarly, $\hat{\mathbf{H}}_{\tau,i}$ denotes the hessian matrix of client i at the end of round τ, and $\hat{\mathbf{H}}_{\tau,i} = H_{\tau,i,K}$ if we have K local epochs. $\tilde{f}_{t,i}(\boldsymbol{\omega})$ approximate $\bar{f}_{t,i}(\boldsymbol{\omega})$, while $\tilde{f}_t(\boldsymbol{\omega})$ approximate $\bar{f}_t(\boldsymbol{\omega})$. When $t = T$, $\tilde{f}_{t,i}(\boldsymbol{\omega})$ is an approximation of $f(\boldsymbol{\omega})$.

There already exists some methods to approximate the Hessian matrix. For classification tasks considered in our numerical evaluation, our approximation follows Kirkpatrick et al. (2017) and uses Fisher Information Matrix.

Information loss. Notice that during the procedure of CFL, the exact Hessian matrix calculation is unavailable and changes over the time. These result in the discrepancy between the estimation and true objective function, which further hinders the optimization. In the following section, we refer it as the information loss, and provide an in-depth discussion.

Definition 3.1 (Information loss)**.** *Suppose $\bar{f}_{t,i}(\boldsymbol{\omega})$ defined in (3), and we use $\tilde{f}_{t,i}(\boldsymbol{\omega})$ defined in (5) to approximate $\bar{f}_{t,i}(\boldsymbol{\omega})$. Then we define the information loss as*

$$\Delta_{t,i}(\boldsymbol{\omega}) = \nabla \bar{f}_{t,i}(\boldsymbol{\omega}) - \nabla \tilde{f}_{t,i}(\boldsymbol{\omega}) .$$

3.3. Distribution drift

Gradient Noise in SGD. In order to analyze the formulated problem, here we introduce the formulation of the distribution drift between clients and rounds. We first recap the standard definition of the gradient noise in Sammut & Webb (2010); Gower et al. (2019).

Definition 3.2 (Gradient noise)**.** *For objective function $f(\boldsymbol{\omega})$, we define the gradient with stochastic noise as*

$$\nabla f(\boldsymbol{\omega}) = g(\boldsymbol{\omega}) + \boldsymbol{\nu} ,$$

where $g(\boldsymbol{\omega})$ is the stochastic gradient, and $\boldsymbol{\nu}$ is a zero-mean noise.

Gradient drift in traditional FL. In FL with heterogeneous client data, the idea of using gradient noise to simulate the distribution drift over clients has been widely used in Karimireddy et al. (2020b); Khaled et al. (2020); Li et al. (2020c), which is formulated below

$$\nabla f(\boldsymbol{\omega}) = \nabla f_i(\boldsymbol{\omega}) + \boldsymbol{\delta}_i ,$$

where $f_i(\boldsymbol{\omega})$ is the local objective function of client i, and $f(\boldsymbol{\omega}) = \sum_{i=1}^{N} p_i f_i(\boldsymbol{\omega})$ is the global objective function we want to optimize.

Gradient drift in FL with Time-Evolving Heterogeneous Data. In this paper, we consider the scenarios of time-evolving heterogeneous data (c.f. Section 3.1), which could be caused by either distribution drift over clients, or inter client drift. Thus, in our formulation (Definition 3.3), we consider both drift between clients and drift over rounds.

Definition 3.3 (Distribution drift)**.** *Considering the drift over both clients and rounds for time-evolving heterogeneous client data, we define*

$$\nabla f(\boldsymbol{\omega}) = \nabla f_{t,i}(\boldsymbol{\omega}) + \boldsymbol{\delta}_{t,i} + \boldsymbol{\xi}_{t,i} ,$$

where $f_{t,i}(\boldsymbol{\omega})$ is the local objective function of client i on round t, $f(\boldsymbol{\omega}) = \mathbb{E}\left[f_{t,i}(\boldsymbol{\omega})\right]$ is the global objective function, $\boldsymbol{\delta}_{t,i}$ is the drift over clients, and $\boldsymbol{\xi}_{t,i}$ is the drift over rounds. Note that $\boldsymbol{\delta}_{t,i}$ remain same for same clients. In scenario S1, clients will only participate in training once, thus, $\boldsymbol{\delta}$ can be independent to each other. However, in other scenarios, different $\boldsymbol{\delta}_{t,i}$ can be correlated.

3.4. Assumptions

This section introduces the assumptions we will use in the theoretical analysis.

Assumption 1 (Smoothness and convexity)**.** *Assume local objective functions f_{ti} are $L-$smooth, and in some conditions, $\mu-$convex. If a function f_{ti} is both $L-$smooth and $\mu-$convex, then it satisfies $\frac{1}{2L} \left\|\nabla f_{t,i}(\mathbf{x}) - \nabla f_{t,i}(\mathbf{y})\right\|^2 \leq f_{t,i}(\mathbf{x}) - f_{t,i}(\mathbf{y}) - \nabla f_{t,i}(\mathbf{x})^T(\mathbf{x} - \mathbf{y})$, and $L \geq \mu$.*

Assumption 2 (Bounded noise in stochastic gradient)**.** *Let $g_{t,i}(\boldsymbol{\omega}) = \nabla f_{t,i}(\boldsymbol{\omega}) + \boldsymbol{\nu}_{t,i,k}$, where $\boldsymbol{\nu}_{t,i,k}$ is the noise in stochastic gradient of client i on round t at k-th local iteration. We assume that $\mathbb{E}\left[\boldsymbol{\nu}|\boldsymbol{\omega}\right] = 0$. In SGD, we use $g_{t,i}(\boldsymbol{\omega})$ instead of $f_{t,i}(\boldsymbol{\omega})$ for gradient descent. We further assume that $\mathbb{E}\left[\left\|\boldsymbol{\nu}\right\|^2 |\boldsymbol{\omega}\right] \leq \sigma^2$.*

Except these two regular assumptions, we also introduce the assumptions to bound the distribution drift over rounds and clients. As shown in Definition 3.3, we have two types of drifts, i.e. drift over clients $\boldsymbol{\delta}_{t,i}$ and drift over rounds $\boldsymbol{\xi}_{t,i}$. To bound this drifts, we have following assumptions.

Assumption 3 (Bounded drift over clients)**.** *Following the Definition 3.3, we assume the bound of client drift $\boldsymbol{\delta}$ as $\mathbb{E}\|\boldsymbol{\delta}\|^2 \leq G^2 + B^2 \mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2$.*

Assumption 4 (Bounded drift over rounds)**.** *Following the Definition 3.3, we assume the bound of client drift $\boldsymbol{\xi}$ as $\mathbb{E}\|\boldsymbol{\xi}\|^2 \leq D^2 + A^2 \mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2$.*

Assumption 5 (Zero mean and independent of drifts)**.** *All $\boldsymbol{\xi}_{t,i}$, $\boldsymbol{\delta}_{t,i}$, and $\boldsymbol{\nu}_{t,i,k}$ are zero mean. $\mathbb{E}\left[\langle \boldsymbol{\xi}, \boldsymbol{\delta} \rangle\right] = 0$, $\mathbb{E}\left[\langle \boldsymbol{\xi}, \boldsymbol{\nu} \rangle\right] = 0$, and $\mathbb{E}\left[\langle \boldsymbol{\delta}, \boldsymbol{\nu} \rangle\right] = 0$ for all $\boldsymbol{\xi}$, $\boldsymbol{\delta}$ and $\boldsymbol{\nu}$.*

Assumption 6 (Independence of client drifts in scenario S1 and S2)**.** *In scenario S1, each client will only appear in training procedure once, thus $\boldsymbol{\delta}_{t,i}$ denotes clients drift of different clients. We assume $\boldsymbol{\delta}_{t,i}$ are independent for all t and i.*
In scenario S2, all clients will be selected in each round, thus $\boldsymbol{\delta}_{t,i}$ denotes client drift of same clients for same i. We assume $\boldsymbol{\delta}_{t,i} = \boldsymbol{\delta}_i$ here.

Assumption 7 (Bounded Hessian dissimilarity)**.** *Denote the hessian matrix of $f_{t,i}(\boldsymbol{\omega})$ as $\mathbf{H}_{t,i}$, and we assume $\|\mathbf{H}_{t_1,i_1} - \mathbf{H}_{t_2,i_2}\| \leq \epsilon$ for any t_1, t_2 and i_1, i_2.*

Note that assumption 7 does not limit the value of ϵ. This assumption can be used to capture how the information loss can affact the optimization.
In previous works (Gower et al., 2019; Karimireddy et al., 2020b), the widely used assumption of distribution drift can be formulated as $\mathbb{E}\|\nabla f_i(\boldsymbol{\omega})\|^2 \leq G^2 + B^2\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2$, where $G^2 \geq 0$ and $B^2 \geq 1$. To match this widely used assumption with our assumptions, we introduce the following lemmas (Proof details refer to Appendix B.3).

Lemma 3.4 (Bound of distribution drift)**.** *Suppose $f_i(\boldsymbol{\omega})$ is the local objective function, and $f(\boldsymbol{\omega}) = \mathbb{E}\left[f_i(\boldsymbol{\omega})\right]$ is the global objective function. Define $\nabla f(\boldsymbol{\omega}) = \nabla f_i(\boldsymbol{\omega}) + \boldsymbol{\varsigma}_i$, $\mathbb{E}[\boldsymbol{\varsigma}_i] = 0$, and assume $\mathbb{E}\left[\|\boldsymbol{\varsigma}\|^2 |\boldsymbol{\omega}\right] \leq A^2\mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + B^2$, we have*

$$\mathbb{E}\left[\|\nabla f_i(\boldsymbol{\omega})\|^2\right] \leq (A^2+1)\mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + B^2\,.$$

We want to compare the performance of CFL-R and FedAvg both in scenario S1 and S2. Notice that FedAvg is a special case of CFL-R (Equation (5)) by setting $p_{t,i} = 1$ on round t. Thus, the local objective function of FedAvg is always $f_{t,i}(\boldsymbol{\omega})$ on round t. Based on Assumption 3-6, Definition 3.3, and Lemma 3.4, we derive the bounded drift of both FedAvg and CFL-R below.

Lemma 3.5 (Bounded drift)**.** *(1) For FedAvg under both scenario S1 and S2, it's local objective function can be bounded as*

$$\mathbb{E}\|\nabla f_{t,i}(\boldsymbol{\omega})\|^2 \leq (1 + A^2 + B^2)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + G^2 + D^2\,.$$

(2) For CFL-R, under scenario S2 where $\boldsymbol{\delta}_{t,i}$ reduces to $\boldsymbol{\delta}_i$, we have

$$\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 \leq \left(1 + B^2 + A^2\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + G^2 + D^2\sum_{\tau=1}^{t} p_{\tau,i}^2\,.$$

(3) For CFL-R, under scenario S1 where $\boldsymbol{\delta}_{t,i}$ are independent for all t and i, we have

$$\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 \leq \left(1 + (B^2 + A^2)\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + (G^2 + D^2)\sum_{\tau=1}^{t} p_{\tau,i}^2\,.$$

The detailed proof of Lemma 3.5 is deferred to Appendix B.2. Lemma 3.5 measures the difference of drift of CFL-R and FedAvg under scenario S1 and S2, and is useful for the further analysis on the convergence rate of both algorithms in Appendix B.4.

4. Theoretical results

In this section, we give the convergence rates of CFL-R (see Algorithm 1), and show that FedAvg (McMahan et al., 2017) is a special case and also a lower bound of our method, for convex objective function.

4.1. Convergence rate of CFL-R

Here we give the convergence rate of CFL-R and FedAvg in scenario S1 and S2 when the objective functions are convex, under Assumptions 7, 1, 2, 3, 4, 5, 6. Table 1 shows the convergence rates of different algorithms compared with prior works.

Theorem 4.1 (Convergence rate of CFL-R under scenario S2)**.** *Suppose our approximation of gradients are accurate ($\|\Delta_{t,i}(\boldsymbol{\omega})\| = 0$, see Definition 3.1), and let $\eta_g \geq 1$.*
For $\mu-$strongly convex $\{f_{ti}\}$, when clients are all selected (scenario S2), the convergence rate of CFL-R is

$$T = \tilde{O}\left(\frac{c_{p_B}}{\mu} + \frac{G^2}{\mu\varepsilon} + \frac{M}{\sqrt{\mu\varepsilon}} + \frac{G}{\mu\sqrt{\varepsilon}} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \sqrt[3]{\frac{D^2}{\mu^2\varepsilon}}\right),$$

and if $\{f_{ti}\}$ are general convex, we have

$$T = \tilde{O}\left(\frac{G^2F^2}{\varepsilon^2} + \frac{(M+c_{p_B})F}{\varepsilon} + \frac{(\sqrt{K}G+\sigma)F^2}{\sqrt{K\varepsilon^3}} + \frac{\sqrt[3]{D^2F^4}}{\varepsilon}\right),$$

where $c_{p_B} = 1 + B^2 + A^2$, $M = \frac{\sigma}{\sqrt{NK}} + D$, and $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$.

We give the proof details in Appendix B.4. Besides, scenario S1 requires the linear independence between $\boldsymbol{\delta}_{t_1,i}$ and $\boldsymbol{\delta}_{t_2,i}$ for all t_1 and t_2, and thus we achieve a tighter convergence rate, as illustrated below.

Theorem 4.2 (Convergence rate of CFL-R under scenario S1)**.** *Suppose our approximation of gradients are accurate ($\|\Delta_{t,i}(\boldsymbol{\omega})\| = 0$), and let $\eta_g \geq 1$.*
For $\mu-$strongly convex $\{f_{ti}\}$,when clients will only appear once (scenario S1), the convergence rate of CFL-R is

$$T = \tilde{O}\left(\frac{c_{p_B}}{\mu} + \frac{M+G}{\sqrt{\mu\varepsilon}} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \sqrt[3]{\frac{G^2+D^2}{\mu^2\varepsilon}}\right),$$

and if $\{f_{ti}\}$ are general convex, we have

$$T = \tilde{O}\left(\frac{(M+G+c_{p_B})F}{\varepsilon} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{\sqrt[3]{(G^2+D^2)F^4}}{\varepsilon}\right),$$

Table 1: **Convergence rates of different algorithms.** Number of communication rounds required to reach ε accuracy for μ strongly convex and general convex functions. (G, B) bounds gradient dissimilar over clients (Assumption 3), and (A, D) bounds gradient dissimilar over rounds (Assumption 4). Our convergence rates for FedAvg and CL match previous works results. Besides, our convergence rate of CL using mini-batch SGD is novel and cover the strongly-convex case.

Algorithm		Strongly Convex	General Convex
SGD		$\frac{\sigma^2}{\mu NK\varepsilon^2} + \frac{1}{\mu}$	$\frac{\sigma^2}{NK\varepsilon^2} + \frac{1}{\varepsilon}$
FedAvg	Li et al. (2020c)	$\frac{\sigma^2}{\mu^2 NK\varepsilon} + \frac{G^2 K}{\mu^2 \varepsilon}$	-
	Khaled et al. (2020)	-	$\frac{G^4+F^4}{N\varepsilon^2}$
	Karimireddy et al. (2020b)	$\frac{W^2}{\mu RK\varepsilon} + \frac{G}{\mu\sqrt{\varepsilon}} + \frac{B^2}{\mu}$	$\frac{W}{KR\varepsilon^2} + \frac{G}{\varepsilon^{\frac{3}{2}}} + \frac{B^2 D^2}{\varepsilon}$
	Ours	$\frac{B^2+1}{\mu} + \frac{\sigma^2}{\mu NK\varepsilon} + \frac{G^2}{\mu\varepsilon} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \frac{G}{\mu\sqrt{K\varepsilon}}$	$\frac{(B^2+1)F}{\varepsilon} + \frac{\sigma^2 F^2}{NK\varepsilon^2} + \frac{G^2 F^2}{\varepsilon^2} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{GF^2}{\sqrt{\varepsilon^3}}$
CL	Yin et al. (2020b)	-	$\frac{1}{\varepsilon}$ (full batch)
	Ours	$\frac{(1+A^2)}{\mu} + \frac{D}{\sqrt{\mu\varepsilon}} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \sqrt[3]{\frac{D^2}{\mu^2\varepsilon}}$	$\frac{\sigma F}{\sqrt{K\varepsilon}} + \frac{(D+1+A^2)F}{\varepsilon} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{\sqrt[3]{D^2F^4}}{\varepsilon}$
CFL-R	Ours	$\frac{c_{p_B}}{\mu} + \frac{G^2}{\mu\varepsilon} + \frac{M_\mu}{\mu\sqrt{\varepsilon}} + \sqrt[3]{\frac{D^2}{\mu^2\varepsilon}}$	$\frac{G^2F^2}{\varepsilon^2} + \frac{(M+c_{p_B})F + \sqrt[3]{D^2F^4}}{\varepsilon} + \frac{(\sqrt{K}G+\sigma)F^2}{\sqrt{K\varepsilon^3}}$

R: number of random sampling clients, $W = \sigma^2 + K(1-\frac{S}{N})G^2$, $M_\mu = \frac{\sigma}{\sqrt{NK}} + D + \frac{G}{\sqrt{\mu}} + \frac{\sigma}{\sqrt{\mu K}}$, $M = \frac{\sigma}{\sqrt{NK}} + D$, $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$, $c_{p_B} = 1 + B^2 + A^2$.

where $c_{p_B} = 1 + B^2 + A^2$, $M = \frac{\sigma}{\sqrt{NK}} + D$, *and* $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$.

FedAvg is a special case of our formulation by setting $p_{\tau,i} = 0$ for $\tau < t$, and $p_{t,i} = 1$. Below, we derive the convergence rate of FedAvg under our new formulation.

Theorem 4.3 (Convergence rate of FedAvg with time-evolving heterogeneous data)**.** *Let $\eta_g \geq 1$. For $\mu-$strongly convex $\{f_{ti}\}$, under scenario S1 or scenario S2, the convergence rate of FedAvg is*

$$T = \tilde{O}\left(\frac{c_{p_B}}{\mu} + \frac{\tilde{M}+G^2}{\mu\varepsilon} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \frac{G+D}{\mu\sqrt{\varepsilon}}\right),$$

and if $\{f_{ti}\}$ are general convex, we have

$$T = \tilde{O}\left(\frac{c_{p_B}F}{\varepsilon} + \frac{(\tilde{M}+G^2)F^2}{\varepsilon^2} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{(G+D)F^2}{\sqrt{\varepsilon^3}}\right),$$

where $c_{p_B} = 1 + B^2 + A^2$, $\tilde{M} = \frac{\sigma^2}{NK} + D^2$, *and* $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$.

Remark 4.4. *The rate of CFL-R outperforms that of FedAvg both in scenario S1 and S2, as shown in the comparison of Theorem 4.3 with Theorem 4.1 and Theorem 4.2. It stems from the fact that CFL-R can reduce the variance of gradient drifts: in scenario S1, CFL-R reduces the variance of round drift $\boldsymbol{\xi}$, whilst in scenario S2, CFL-R reduces both round drift $\boldsymbol{\xi}$ and client drift $\boldsymbol{\delta}$. The detail refers to Appendix B.4.*

4.2. Connection with prior work

Our CFL-R framework covers conventional scenarios in both federated learning and continual leaning.

Convergence of FedAvg in Traditional FL scenario. By setting $A = 0, D = 0$ in Assumption 4, we recover the convergence rate of FedAvg in traditional FL scenario:

Theorem 4.5 (Convergence rate of FedAvg)**.** *Let $A = 0, D = 0$ and $\eta_g = 1$ in Assumption 4, and assume $\{f_{ti}\}$ are $\mu-$strongly convex. Then, if all clients are selected (scenario 1), the convergence rate of FedAvg is derived as*

$$T = \tilde{O}\left(\frac{B^2+1}{\mu} + \frac{\sigma^2}{\mu NK\varepsilon} + \frac{G^2}{\mu\varepsilon} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \frac{G}{\mu\sqrt{K\varepsilon}}\right),$$

and when $\{f_{ti}\}$ are general convex, we have

$$T = \tilde{O}\left(\frac{(B^2+1)F}{\varepsilon} + \frac{\sigma^2F^2}{NK\varepsilon^2} + \frac{G^2F^2}{\varepsilon^2} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{GF^2}{\sqrt{\varepsilon^3}}\right),$$

where $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$.

As shown in Table 1, our convergence rate of FedAvg in traditional FL scenario match previous works' analysis. In particular, our results is almost identical to Karimireddy et al. (2020b) if remove the random client selection.

Convergence rate of CL in traditional CL scenario. Our formulation of CFL-R can be reduced to the standard continual learning setup by simply setting $G = 0, B = 0$ in Assumption 3, and set client number $N = 1$. Then we derive the convergence rate of standard continual learning.

Theorem 4.6 (Convergence rate of standard continual learning)**.** *Suppose our approximation of gradients are accurate ($\|\Delta_{t,i}(\boldsymbol{\omega})\| = 0$).*
For $\mu-$strongly convex $\{f_{ti}\}$, the convergence rate of the standard continual learning can be derived as below.

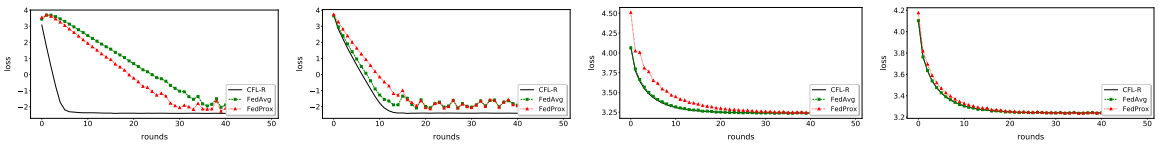
$$T = \tilde{O}\left(\frac{(1+A^2)}{\mu} + \frac{\sigma}{\sqrt{\mu K\varepsilon}} + \frac{D}{\sqrt{\mu\varepsilon}} + \frac{\sigma}{\mu\sqrt{K\varepsilon}} + \sqrt[3]{\frac{D^2}{\mu^2\varepsilon}}\right),$$

and when $\{f_{ti}\}$ are general convex, we have

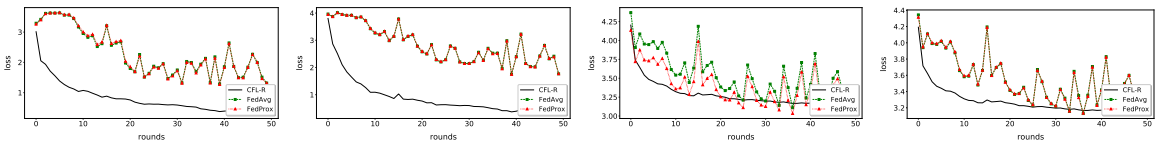
$$T = \tilde{O}\left(\frac{\sigma F}{\sqrt{K\varepsilon}} + \frac{(D+1+A^2)F}{\varepsilon} + \frac{\sigma F^2}{\sqrt{K\varepsilon^3}} + \frac{\sqrt[3]{D^2F^4}}{\varepsilon}\right),$$

where $F = \mathbb{E}\left[\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^\star\|\right]$.

We are the first to prove the convergence rate by considering mini batch SGD in CL scenario, whereas the existing work,



(a) Small L, strongly convex, small round drift (b) Large L, strongly convex, small round drift (c) Small L, general convex, small round drift (d) Large L, general convex, small round drift



(e) Small L, strongly convex, big round drift (f) Large L, strongly convex, big round drift (g) Small L, general convex, big round drift (h) Large L, general convex, big round drift

Figure 1: Performance of different algorithms on noisy quadratic model

e.g., Yin et al. (2020b), only considers the full batch gradient descent for general convex objective functions. Note that setting $D = 0, A = 0$ in Assumption 4 and $\sigma = 0$ in Assumption 2 recover gradient descent—it gives the convergence rate of $\tilde{O}(\frac{1}{T})$ for general convex functions, as in Yin et al. (2020b). In addition, we also give the convergence rate for strongly convex functions.

5. Practical Implementation of CFL-R

Lemma 3.5 indicates that the bound of drift of CFL-R is related to the value of $\left(1 + B^2 + A^2 \sum_{\tau=1}^{t} p_{\tau,i}^2\right)$ and $\left(G^2 + D^2 \sum_{\tau=1}^{t} p_{\tau,i}^2\right)$. We can set $p_{\tau,i} = \frac{1}{t}$ to minimize this drift. This observation leads to a naive CFL implementation, shown in Algorithm 2 of Appendix C.2.

In Algorithm 2, at the beginning of round t, the server not only transmits model parameters to local clients, but also $\mathbf{H}_{agg,i} = \sum_{\tau=1}^{t-1} p_{\tau,i} \hat{\mathbf{H}}_{\tau,i}$, and $\mathbf{G}_{agg,i} = \sum_{\tau=1}^{t-1} p_{\tau,i} \left(\hat{\mathbf{G}}_{\tau,i} - \hat{\mathbf{H}}_{\tau,i} \hat{\boldsymbol{\omega}}_{\tau,i}\right)$. Here $\hat{\mathbf{H}}_{\tau,i}$ is the Hessian matrix evaluated on the parameters at the end of round τ on client i, $\hat{\mathbf{G}}_{\tau,i}$ corresponds to the gradients, and $\hat{\boldsymbol{\omega}}_{\tau,i}$ is the model parameter. We use $\mathbf{H}_{agg,i} \hat{\boldsymbol{\omega}}_{\tau,i} + \mathbf{G}_{agg,i}$ to approximate $\sum_{\tau=1}^{t-1} p_{\tau,i} \nabla f_{\tau,i}(\hat{\boldsymbol{\omega}}_{\tau,i})$.

Because $p_{\tau,i} = \frac{1}{t}$ in Algorithm 2, we have $\mathbf{H}_{agg,i} = \frac{1}{t} \sum_{\tau=1}^{t-1} \hat{\mathbf{H}}_{\tau,i}$, and $\mathbf{G}_{agg,i} = \frac{1}{t} \sum_{\tau=1}^{t-1} \left(\hat{\mathbf{G}}_{\tau,i} - \hat{\mathbf{H}}_{\tau,i} \hat{\boldsymbol{\omega}}_{\tau,i}\right)$. Each local update step of the Exact CFL-R uses the following update rule

$$\boldsymbol{\omega}_{t,i,k+1} = \boldsymbol{\omega}_{t,i,k} - \eta_l \left(\frac{1}{t} g_{t,i}(\boldsymbol{\omega}_{t,i,k}) + \mathbf{G}_{agg,i} + \mathbf{H}_{agg,i} \boldsymbol{\omega}_{t,i,k}\right).$$

At the end of local training (i.e., after K local iterations), clients calculate the Hessians using local parameters $\boldsymbol{\omega}_{t,i,K}$ and local data, and upload the final parameters, Hessians and gradients to the server for the further aggregation.

6. Experiments

Following the idea proposed in Zhang et al. (2019), we use the noisy quadratic model, $F(\boldsymbol{\omega}) = \boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} + \mathbf{B}^T \boldsymbol{\omega} + C$, to construct a FL scenario that captures all our assumptions. $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{A} \succeq 0$, $\mathbf{B} \in \mathbb{R}^n$, $C \in \mathbb{R}$ are the model parameters. To simulate (local) SGD with distribution drift, we add stochastic noise to the true gradient $\mathbf{A}\boldsymbol{\omega} + \mathbf{B}$. For local client i on round t, we generate $g_{t,i,k}(\boldsymbol{\omega}) = \mathbf{A}\boldsymbol{\omega} + \mathbf{B} + \boldsymbol{\delta}_i + \boldsymbol{\xi}_{t,i} + \boldsymbol{\nu}_{t,i,k}$ as formulated in Definition 3.3 and Assumption 2. All drifts and noise are generated from Normal distribution with zero mean and different variances. See Section C.1 for more details.

We compare exact-CFL-R (Algorithm 2) with other baselines, like FedAvg (McMahan et al., 2017), FedProx (Li et al., 2018). We perform gradient descent with a constant learning rate (tuned for each setting). For simplicity, we report the $\|A\boldsymbol{\omega} + B\|$.

As shown in Figure 1, our CFL-R algorithm outperforms all other methods, in all evaluated settings. For strongly convex problem or under the case of large round drift, CFL-R significantly outperforms FedAvg and FedProx. For large round drifts, the learning curves of FedAvg and FedProx illustrate severe oscillation, while CFL-R has a much smoother curve in all settings. It is aligned with Lemma 3.5 that CFL-R has the effect of variance reduction (on gradients): when uniformly weighted, the variance will be reduced for t times on round t.

7. Conclusion

Our work introduced a new analysis framework, Continual Federated Learning (CFL), to formulate time-evolving heterogeneous data in FL, and also integrate the formulations of both classical FL and CL. We proposed a new regularization-based algorithm CFL-R, and prove that the convergence rate of CFL-R is better than FedAvg. Besides, we give a novel convergence rate by considering mini-batch

SGD in CL scenario.
We compare the performance of CFL-R, FedAvg, and FedNova using noisy quadratic model, and verified that CFL-R converges faster than other FL baselines.

8. Acknowledgement

This work is supported in part by the funding from Shenzhen Institute of Artificial Intelligence and Robotics for Society, the National Key R&D Program of China with grant No. 2018YFB1800800, and the National Natural Science Foundation of China (NSFC) under Grant No. 62001412.

References

Bui, T. D., Nguyen, C. V., Swaroop, S., and Turner, R. E. Partitioned variational inference: A unified framework encompassing federated and continual learning. *arXiv preprint arXiv:1811.11206*, 2018.

Castro, F. M., Marín-Jiménez, M. J., Guil, N., Schmid, C., and Alahari, K. End-to-end incremental learning. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 233–248, 2018.

Das, R., Acharya, A., Hashemi, A., Sanghavi, S., Dhillon, I. S., and Topcu, U. Faster non-convex federated learning via global and local momentum. *arXiv preprint arXiv:2012.04061*, 2020.

Diao, E., Ding, J., and Tarokh, V. Heterofl: Computation and communication efficient federated learning for heterogeneous clients. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=TNkPBBYFkXg.

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

French, R. M. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

Gower, R. M., Loizou, N., Qian, X., Sailanbayev, A., Shulgin, E., and Richtárik, P. Sgd: General analysis and improved rates. In *International Conference on Machine Learning*, pp. 5200–5209. PMLR, 2019.

Haddadpour, F., Kamani, M. M., Mokhtari, A., and Mahdavi, M. Federated learning with compression: Unified analysis and sharp guarantees. In *International Conference on Artificial Intelligence and Statistics*, pp. 2350–2358. PMLR, 2021.

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

Karimireddy, S. P., Jaggi, M., Kale, S., Mohri, M., Reddi, S. J., Stich, S. U., and Suresh, A. T. Mime: Mimicking centralized stochastic algorithms in federated learning. *arXiv preprint arXiv:2008.03606*, 2020a.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pp. 5132–5143. PMLR, 2020b.

Khaled, A., Mishchenko, K., and Richtárik, P. Tighter theory for local sgd on identical and heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pp. 4519–4529. PMLR, 2020.

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

Koloskova, A., Loizou, N., Boreiri, S., Jaggi, M., and Stich, S. A unified theory of decentralized sgd with changing topology and local updates. In *International Conference on Machine Learning*, pp. 5381–5393. PMLR, 2020.

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. *arXiv preprint arXiv:1812.06127*, 2018.

Li, T., Hu, S., Beirami, A., and Smith, V. Ditto: Fair and robust federated learning through personalization. *arXiv preprint arXiv:2012.04221*, 2020a.

Li, T., Sanjabi, M., Beirami, A., and Smith, V. Fair resource allocation in federated learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020b. URL https://openreview.net/forum?id=ByexElSYDr.

Li, X., Huang, K., Yang, W., Wang, S., and Zhang, Z. On the convergence of fedavg on non-iid data. In *International Conference on Learning Representations*, 2020c. URL https://openreview.net/forum?id=HJxNAnVtDS.

Li, Z. and Hoiem, D. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

Lin, T., Stich, S. U., Patel, K. K., and Jaggi, M. Don't use large mini-batches, use local sgd. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1eyO1BFPr.

Maltoni, D. and Lomonaco, V. Continuous learning in single-incremental-task scenarios. *Neural Networks*, 116: 56–73, 2019.

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

Mitra, A., Jaafar, R., Pappas, G. J., and Hassani, H. Achieving linear convergence in federated learning under objective and systems heterogeneity. *arXiv preprint arXiv:2102.07053*, 2021.

Mohri, M., Sivek, G., and Suresh, A. T. Agnostic federated learning. In Chaudhuri, K. and Salakhutdinov,

R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4615–4625. PMLR, 2019. URL http://proceedings.mlr.press/v97/mohri19a.html.

Patel, K. K. and Dieuleveut, A. Communication trade-offs for synchronized distributed sgd with large step size. In *Advances in Neural Information Processing Systems*, 2019.

Rebuffi, S.-A., Kolesnikov, A., Sperl, G., and Lampert, C. H. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pp. 2001–2010, 2017.

Reisizadeh, A., Farnia, F., Pedarsani, R., and Jadbabaie, A. Robust federated learning: The case of affine distribution shifts. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020. URL https://proceedings.neurips.cc/paper/2020/hash/f5e536083a438cec5b64a4954abc17f1-Abstract.html.

Rusu, A. A., Rabinowitz, N. C., Desjardins, G., Soyer, H., Kirkpatrick, J., Kavukcuoglu, K., Pascanu, R., and Hadsell, R. Progressive neural networks. *arXiv preprint arXiv:1606.04671*, 2016.

Sammut, C. and Webb, G. I. (eds.). *Neuro-Dynamic Programming*, pp. 716–716. Springer US, Boston, MA, 2010. ISBN 978-0-387-30164-8. doi: 10.1007/978-0-387-30164-8_588. URL https://doi.org/10.1007/978-0-387-30164-8_588.

Stich, S. U. Local SGD converges fast and communicates little. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=S1g2JnRcFX.

Stich, S. U. and Karimireddy, S. P. The error-feedback framework: Better rates for sgd with delayed gradients and compressed updates. *Journal of Machine Learning Research*, 21:1–36, 2020.

Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. Tackling the objective inconsistency problem in heterogeneous federated optimization. *arXiv preprint arXiv:2007.07481*, 2020.

Woodworth, B., Patel, K. K., and Srebro, N. Minibatch vs local sgd for heterogeneous distributed learning. In *NeurIPS 2020 - Thirty-fourth Conference on Neural Information Processing Systems*, 2020a.

Woodworth, B., Patel, K. K., Stich, S., Dai, Z., Bullins, B., Mcmahan, B., Shamir, O., and Srebro, N. Is local sgd better than minibatch sgd? In *International Conference on Machine Learning*, pp. 10334–10343. PMLR, 2020b.

Yang, H., Fang, M., and Liu, J. Achieving linear speedup with partial worker participation in non-iid federated learning. *arXiv preprint arXiv:2101.11203*, 2021.

Yin, D., Farajtabar, M., and Li, A. SOLA: continual learning with second-order loss approximation. *CoRR*, abs/2006.10974, 2020a. URL https://arxiv.org/abs/2006.10974.

Yin, D., Farajtabar, M., Li, A., Levine, N., and Mott, A. Optimization and generalization of regularization-based continual learning: a loss approximation viewpoint. *arXiv preprint arXiv:2006.10974*, 2020b.

Yoon, J., Jeong, W., Lee, G., Yang, E., and Hwang, S. J. Federated continual learning with adaptive parameter communication. *CoRR*, abs/2003.03196, 2020. URL https://arxiv.org/abs/2003.03196.

Yu, H., Jin, R., and Yang, S. On the linear speedup analysis of communication efficient momentum SGD for distributed non-convex optimization. In Chaudhuri, K. and Salakhutdinov, R. (eds.), *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7184–7193. PMLR, 2019. URL http://proceedings.mlr.press/v97/yu19d.html.

Zenke, F., Poole, B., and Ganguli, S. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pp. 3987–3995. PMLR, 2017.

Zhang, G., Li, L., Nado, Z., Martens, J., Sachdeva, S., Dahl, G. E., Shallue, C. J., and Grosse, R. Which algorithmic choices matter at which batch sizes? insights from a noisy quadratic model. In *Advances in Neural Information Processing Systems*, 2019.

Zinkevich, M., Weimer, M., Smola, A. J., and Li, L. Parallelized stochastic gradient descent. In *NIPS*, volume 4, pp. 4. Citeseer, 2010.

Contents of Appendix

A. Techniques

Here we show some technical lemmas which are useful in the proof part.

Lemma A.1 (linear convergence rate ((Karimireddy et al., 2020b), Lemma 1)). *For every non-negative sequence $\{d_{r-1}\}_{r\geq 1}$, and any parameter $\mu > 0$, $T \geq \frac{1}{2\eta_{max}\mu}$, there exists a constant step-size $\eta \leq \eta_{max}$ and weights $\omega_t = (1-\mu\eta)^{1-t}$ such that for $W_T = \sum_{t=1}^{T+1} \omega_t$*

$$\Phi_T = \frac{1}{W_T}\sum_{t=1}^{T+1}\left(\frac{\omega_{t-1}}{\eta}(1-\mu\eta)d_{t-1} - \frac{\omega_t}{\eta}d_t\right) = \tilde{\mathcal{O}}\left(\mu d_0 exp(-\mu\eta_{max}T)\right) . \tag{7}$$

Lemma A.2 (sub-linear convergence rate ((Karimireddy et al., 2020b), Lemma 2)). *For every non-negative sequence $\{d_{r-1}\}_{r\geq 1}$ and any parameters $\eta_{max} > 0$, $c_1 \geq 0$, $c_2 \geq 0$, $T \geq 0$, there exists a constant step-size $\eta \leq \eta_{max}$ such that*

$$\Phi_T = \frac{1}{T+1}\sum_{t=1}^{T+1}\left(\frac{d_{t-1}}{\eta} - \frac{d_t}{\eta} + c_1\eta + c_2\eta^2\right) \leq \frac{d_0}{\eta_{max}(T+1)} + \frac{2\sqrt{c_1 d_0}}{\sqrt{T+1}} + 2\left(\frac{d_0}{T+1}\right)^{\frac{2}{3}} c_2^{\frac{1}{3}} . \tag{8}$$

Lemma A.3 (relaxed triangle inequality ((Karimireddy et al., 2020b), Lemma 3)). *Let $\{\mathbf{v}_1, ..., \mathbf{v}_\tau\}$ be τ vectors in $\mathcal{R}^d$. Then the following is true:*

$$\|\frac{1}{\tau}\sum_{i=1}^{\tau}\mathbf{v}_i\|^2 \leq \frac{1}{\tau}\sum_{i=1}^{\tau}\|\mathbf{v}_i\|^2 . \tag{9}$$

Lemma A.4 (separating mean and variance((Stich & Karimireddy, 2020), Lemma 14)). *Let $\Xi_1, \Xi_2, ..., \Xi_\tau$ be τ random variables in $\mathcal{R}^d$ which are not necessarily independent. First suppose that their mean is $E[\Xi_i] = \xi_i$, and variance is bounded as $\mathbb{E}\|\Xi_i - \xi_i\|^2 \leq M\|\xi_i\|^2 + \sigma^2$, then the following holds*

$$\mathbb{E}\left\|\sum_{i=1}^{\tau}\Xi_i\right\|^2 \leq (\tau + M)\sum_{i=1}^{\tau}\|\xi_i\|^2 + \tau\sigma^2 . \tag{10}$$

Lemma A.5 (perturbed strongly convexity((Karimireddy et al., 2020b), Lemma 5)). *The following holds for any L-smooth and μ-strongly convex function h, and any $\mathbf{x}$, $\mathbf{y}$, $\mathbf{z}$ in the domain of h:*

$$\langle\nabla h(\mathbf{x}), \mathbf{z} - \mathbf{y}\rangle \geq h(\mathbf{z}) - h(\mathbf{y}) + \frac{\mu}{4}\|\mathbf{y} - \mathbf{z}\|^2 - L\|\mathbf{z} - \mathbf{x}\|^2 . \tag{11}$$

Lemma A.6. *Define $S = \sum_{t=1}^{T}\frac{p_t}{t}$ where $\sum_{t=1}^{T} p_t = 1$, and $p_t \leq p_{t+1}$ then $S \leq \frac{1}{T}\sum_{t=1}^{T}\frac{1}{t} = \tilde{\mathcal{O}}\left(\frac{lnT}{T}\right)$.*

B. Convergence rate of CFL-R

B.1. Proof of Lemma 3.4

Lemma B.1 (Bound of distribution drift)**.** *Suppose $f_i(\boldsymbol{\omega})$ is the local objective function, and $f(\boldsymbol{\omega}) = \mathbb{E}[f_i(\boldsymbol{\omega})]$ is the global objective function. Define $\nabla f(\boldsymbol{\omega}) = \nabla f_i(\boldsymbol{\omega}) + \boldsymbol{\varsigma}_i$, $\mathbb{E}[\boldsymbol{\varsigma}_i] = 0$, and assume $\mathbb{E}\left[\|\boldsymbol{\varsigma}\|^2 |\boldsymbol{\omega}\right] \leq A^2\mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + B^2$, we have*

$$\mathbb{E}\left[\|\nabla f_i(\boldsymbol{\omega})\|^2\right] \leq (A^2+1)\mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + B^2\,. \tag{12}$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[\|\nabla f_i(\boldsymbol{\omega})\|^2\right] &= \mathbb{E}\left[\|\nabla f(\boldsymbol{\omega}) + \boldsymbol{\varsigma}_i\|^2\right] && (13)\\
&= \mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + \mathbb{E}\left[\|\boldsymbol{\varsigma}_i\|^2\right] && (14)\\
&\leq (A+1)\mathbb{E}\left[\|\nabla f(\boldsymbol{\omega})\|^2\right] + B\,. && (15)
\end{aligned}$$

□

B.2. Proof of Lemma 3.5

Lemma B.2 (Bounded drift)**.** *For formulation (5), consider CFL-R and FedAvg in scenario S1 and scenario S2, we can bound the gradient drift as*
(1) FedAvg only optimize current local objective functions, thus both in scenario S1 and S2,

$$\mathbb{E}\|\nabla f_{t,i}(\boldsymbol{\omega})\|^2 \leq (1 + A^2 + B^2)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + G^2 + D^2\,.$$

(2) For CFL-R, in scenario S2 ($\boldsymbol{\delta}_{t,i} = \boldsymbol{\delta}_i$), we have

$$\begin{aligned}
\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 \leq &\left(1 + B^2 + A^2\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2\\
&+ G^2 + D^2\sum_{\tau=1}^{t} p_{\tau,i}^2\,.
\end{aligned}$$

(3) For CFL-R, in scenario S1($\boldsymbol{\delta}_{t,i}$ are independent for all t and i), we have

$$\begin{aligned}
\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 \leq &\left(1 + (B^2 + A^2)\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2\\
&+ (G^2 + D^2)\sum_{\tau=1}^{t} p_{\tau,i}^2\,.
\end{aligned}$$

Proof. Using Assumption 3 and 4, together with the fact that $\nabla f_{t,i}(\boldsymbol{\omega}) = \nabla f(\boldsymbol{\omega}) + \boldsymbol{\delta}_{t,i} + \boldsymbol{\xi}_{t,i}$, we have

$$\begin{aligned}
\mathbb{E}\|\nabla f_{t,i}(\boldsymbol{\omega})\|^2 &= \mathbb{E}\left\|\nabla f(\boldsymbol{\omega}) + \boldsymbol{\delta}_{t,i} + \boldsymbol{\xi}_{t,i}\right\|^2 && (16)\\
&= \mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + \mathbb{E}\|\boldsymbol{\delta}_{t,i}\|^2 + \mathbb{E}\left\|\boldsymbol{\xi}_{t,i}\right\|^2 && (17)\\
&\leq (1 + A^2 + B^2)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + G^2 + D^2\,. && (18)
\end{aligned}$$

The second inequality is based on Assumption 5, and directly use Assumption 3 and Assumption 4 we get the last inequality. Similarly, we have

$$\begin{aligned}
\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 &= \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}) + \boldsymbol{\delta}_i + \boldsymbol{\xi}_{\tau,i})\right\|^2 && (19)\\
&= \mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + \mathbb{E}\|\boldsymbol{\delta}_i\|^2 + \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\boldsymbol{\xi}_{\tau,i}\right\|^2 && (20)\\
&\leq \left(1 + A^2 + B^2\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + G^2 + D^2\sum_{\tau=1}^{t} p_{\tau,i}^2\,. && (21)
\end{aligned}$$

For the last inequality, when $\boldsymbol{\delta}_{t_1,i}$ and $\boldsymbol{\delta}_{t_2,i}$ are independent for all t_1, t_2,

$$\mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 = \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}) + \boldsymbol{\delta}_{t,i} + \boldsymbol{\xi}_{\tau,i})\right\|^2 \tag{22}$$

$$= \mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\boldsymbol{\delta}_{t,i}\right\|^2 + \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\boldsymbol{\xi}_{\tau,i}\right\|^2 \tag{23}$$

$$\leq \left(1 + (A^2 + B^2)\sum_{\tau=1}^{t} p_{\tau,i}^2\right)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 + (G^2 + D^2)\sum_{\tau=1}^{t} p_{\tau,i}^2 . \tag{24}$$

□

B.3. Bouned gradient dissimilarity of objective functions

Lemma B.3 (Bounded gradient dissimilarity of objective functions)**.** *For $\Delta_{t,i}(\boldsymbol{\omega}) = \nabla\bar{f}_{t,i}(\boldsymbol{\omega}) - \nabla\tilde{f}_{t,i}(\boldsymbol{\omega})$ (c.f. Definition 3.1), we have*

$$\|\Delta_{t,i}(\boldsymbol{\omega})\| \leq \frac{\epsilon}{\sum_{\tau=1}^{t} p_\tau}\sum_{\tau=1}^{t-1} p_\tau \left\|\boldsymbol{\omega} - \hat{\boldsymbol{\omega}}_{\tau,i}\right\| .$$

Proof.

$$\|\Delta_{t,i}(\boldsymbol{\omega})\| = \left\|\nabla\bar{f}_{t,i}(\boldsymbol{\omega}) - \nabla\tilde{f}_{t,i}(\boldsymbol{\omega})\right\| \tag{25}$$

$$= \left\|\sum_{\tau=1}^{t-1} p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}) - \nabla f_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K}) - \mathbf{H}_{tjK}(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}))\right\| \tag{26}$$

$$= \left\|\sum_{\tau=1}^{t-1} p_{\tau,i}(\nabla^2 f_{\tau,i}(\boldsymbol{\omega}_{\boldsymbol{\xi},t})(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}) - \mathbf{H}_{tjK}(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}))\right\| \tag{27}$$

$$= \left\|\sum_{\tau=1}^{t-1} p_{\tau,i}((\nabla^2 f_{\tau,i}(\boldsymbol{\omega}_{\boldsymbol{\xi},t}) - \mathbf{H}_{tjK})(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}))\right\| \tag{28}$$

$$\leq \frac{\epsilon}{\sum_{\tau=1}^{t} p_{\tau,i}}\sum_{\tau=1}^{t-1} p_{\tau,i}\|\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}\| . \tag{29}$$

The first two equations come from the mean value theorem which says for continuous function f in closed intervals $[a, b]$ and differentiable on open interval (a, b), there exists a point $c \subseteq (a, b)$ such that

$$f'(c) = \frac{f(b) - f(a)}{b - a}. \tag{30}$$

The last inequality is based on Assumption 4. □

B.4. Proof of Theorem 4.2

In this section, we will give the complete proof of convergence rate of algorithm 1 when the objective functions are convex. As shown in the algorithm, in each round, server broadcast gradients and Hessians of previous rounds, and after local iterations, calculate Hessians and gradients using current local parameters and datasets, then upload them to server.
The local objective function of round t on client i is

$$\tilde{f}_{t,i}(\boldsymbol{\omega}) = (p_{t,i}f_{t,i}(\boldsymbol{\omega}) + \sum_{\tau=1}^{t-1} p_{\tau,i}(f_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K}) + \nabla f_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K})^T(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}) + \frac{1}{2}(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K})^T\mathbf{H}_{\tau,i}(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}))) . \tag{31}$$

Without loss of generality, assume $\sum_{\tau=1}^{t} p_{\tau,i} = 1$. Then follow the steps in Algorithm 2, we have

$$\nabla\tilde{f}_{t,i}(\boldsymbol{\omega}) = (p_{t,i}\nabla f_{t,i}(\boldsymbol{\omega}) + \sum_{\tau=1}^{t-1} p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K}) + \mathbf{H}_{\tau,i}(\boldsymbol{\omega} - \boldsymbol{\omega}_{\tau,i,K}))) . \tag{32}$$

Algorithm 1 CFL-R Framework

Require: initial weights $\boldsymbol{\omega}_0$, global learning rate η_g, local learning rate η_l

for round t = 1,...,T **do**
  for client i = 1,...,N **do**
    $\mathbf{H}_{agg,i} \leftarrow \sum_{\tau=1}^{t-1} p_{\tau,i}\mathbf{H}_{\tau,i}$.
    $\mathbf{G}_{agg,i} \leftarrow \sum_{\tau=1}^{t-1} p_{\tau,i}(\mathbf{G}_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K}) - \mathbf{H}_{\tau,i}\boldsymbol{\omega}_{\tau,i,K})$.
  communicate $\boldsymbol{\omega}_t$, $\mathbf{H}_{agg,i}$, and $\mathbf{G}_{agg,i}$ to client j.
  for client i = 1,...,N **in parallel do**
    initialize local model $\boldsymbol{\omega}_{t,i,0} = \boldsymbol{\omega}_t$.
    for k = 1,...,K **do**
      compute mini-batch gradient $g_{t,i,k}(\boldsymbol{\omega}_{t,i,k-1})$.
      $\boldsymbol{\omega}_{t,i,k} \leftarrow \boldsymbol{\omega}_{t,i,k-1} - \eta_l(p_{t,i}g_{t,i,k}(\boldsymbol{\omega}_{t,i,k-1}) + \mathbf{G}_{agg,i} + \mathbf{H}_{agg,i}\boldsymbol{\omega}_{t,i,k-1})$.
    $\mathbf{H}_{t,i}, \mathbf{G}_{t,i} \leftarrow calculateHessianAndGradient(\boldsymbol{\omega}_{t,i,K})$.
    communicate $\Delta\boldsymbol{\omega}_{t,i} \leftarrow \boldsymbol{\omega}_{t,i,K} - \boldsymbol{\omega}_t$, $\mathbf{H}_{t,i}$, $\mathbf{G}_{t,i}$.
  $\Delta\boldsymbol{\omega}_t \leftarrow \frac{\eta_g}{N}\sum_{i=1}^{N}\Delta\boldsymbol{\omega}_{t,i}$.
  $\boldsymbol{\omega}_{t+1} \leftarrow \boldsymbol{\omega}_t + \Delta\boldsymbol{\omega}_t$.

In mini-batch SGD, we can't get $\nabla f_{t,i}(\boldsymbol{\omega})$ directly, instead, we can only use the noisy gradient $g_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) = \nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) + \boldsymbol{\nu}_{t,i,k}$(Assumption 2), where $\boldsymbol{\nu}_{t,i,k}$ is the noise of mini-batch SGD. Then follow the steps in algorithm 2, we have

$$\tilde{g}_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) = \nabla\tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) + \sum_{\tau=1}^{t} p_{\tau,i}\boldsymbol{\nu}_{t,j,k}\,, \tag{33}$$

$$\Delta\boldsymbol{\omega}_t = \frac{-\eta_l\eta_g}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\tilde{g}_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\,, \tag{34}$$

$$E[\Delta\boldsymbol{\omega}_t] = \frac{-\eta_l\eta_g}{N}\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla\tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\,. \tag{35}$$

By setting $\eta = K\eta_l\eta_g$, we have

$$\Delta\boldsymbol{\omega}_t = \frac{-\eta}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\tilde{g}_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\,, \tag{36}$$

$$E[\Delta\boldsymbol{\omega}_t] = \frac{-\eta}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\nabla\tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\,. \tag{37}$$

Then based on the one-step progress,

$$\mathbb{E}\|\boldsymbol{\omega}_t + \Delta\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 = \|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 - \underbrace{2E\langle\boldsymbol{\omega}_t - \boldsymbol{\omega}^*, \Delta\boldsymbol{\omega}_t\rangle}_{A_1} + \underbrace{\mathbb{E}\|\Delta\boldsymbol{\omega}_t\|^2}_{A_2}\,. \tag{38}$$

Firstly consider A_1 part in equation (38), and let $\bar{f}_{t,i}(\boldsymbol{\omega}) = \sum_{\tau=1}^{t} p_{\tau,i} f_{\tau,i}(\boldsymbol{\omega})$, follow the Lemma B.3, we have

$$
\begin{aligned}
& -2E\langle \boldsymbol{\omega}_t - \boldsymbol{\omega}^*, \Delta\boldsymbol{\omega}_t \rangle \\
&= \frac{2\eta}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \langle \nabla \tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}^* - \boldsymbol{\omega}_t \rangle && (39) \\
&= \frac{2\eta}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \langle \nabla \bar{f}_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) - \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}^* - \boldsymbol{\omega}_t \rangle && (40) \\
&= \frac{2\eta}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\tau=1}^{t} p_{\tau,i} \langle \nabla f_{\tau,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}^* - \boldsymbol{\omega}_t \rangle + \frac{2\eta}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \langle \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}_t - \boldsymbol{\omega}^* \rangle && (41) \\
&\leq \frac{2\eta}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\tau=1}^{t} p_{\tau,i} \langle \nabla f_{\tau,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}^* - \boldsymbol{\omega}_t \rangle + \frac{2\eta\epsilon}{NK \sum_{\tau=1}^{t} p_{\tau,i}} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\tau=1}^{t-1} p_{\tau,i} \|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_{\tau,i,K}\| \|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\| . && (42)
\end{aligned}
$$

According to Equation (37), we derive the first inequality and directly use Lemma B.3 and Cauchy–Schwarz inequality for the last inequality.
From the above inequality, we notice that when the difference of hessian ϵ is close to zero, it is as same as optimizing the true objective function. However, when the difference of hessian becomes high, the performance drops.
Then we can consider the first term of inequality (42). Firstly, by Lemma A.5,

$$
\langle \nabla f_{\tau,i}(\boldsymbol{\omega}_{t,i,k-1}), \boldsymbol{\omega}^* - \boldsymbol{\omega}_t \rangle \leq f_{\tau,i}(\boldsymbol{\omega}^*) - f_{\tau,i}(\boldsymbol{\omega}_t) - \frac{\mu}{4} \|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 + L \|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\|^2 . \tag{43}
$$

Combine (42) and (43), we have,

$$
\begin{aligned}
& -2E\langle \boldsymbol{\omega}_t - \boldsymbol{\omega}^*, \Delta\boldsymbol{\omega}_t \rangle \\
&\leq -2\eta (f_t(\boldsymbol{\omega}_t) - f_t(\boldsymbol{\omega}^*) + \frac{\mu}{4} \|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2) + \frac{2\eta L}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\|^2 \\
&+ \frac{2\eta\epsilon}{NK} \sum_{i=1}^{N} \sum_{k=1}^{K} \sum_{\tau=1}^{t-1} p_{\tau,i} \|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\| \|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\| . && (44)
\end{aligned}
$$

Next, consider A_2 in (38). According to Lemma 3.5,

$$\mathbb{E}\|\Delta\boldsymbol{\omega}_t\|^2$$

$$= \frac{\eta^2}{N^2K^2}\mathbb{E}\left\|\sum_{i=1}^{N}\sum_{k=1}^{K}(p_{t,i}\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) + \sum_{\tau=1}^{t-1}p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}_t) + H_{t,j}(\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t)) + \sum_{\tau=1}^{t}p_{\tau,i}\boldsymbol{\nu}_{tjk})\right\|^2 \tag{45}$$

$$\leq \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\mathbb{E}\left\|p_{t,i}\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) + \sum_{\tau=1}^{t-1}p_{\tau,i}(\nabla f_{\tau,i}(\boldsymbol{\omega}_t) + H_{t,j}(\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t))\right\|^2 + \frac{\eta^2\sum_{i=1}^{N}\sum_{\tau=1}^{t}p_{\tau,i}^2\sigma^2}{N^2K} \tag{46}$$

$$= \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\mathbb{E}\left\|\sum_{\tau=1}^{t}p_{\tau,i}\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}) - \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\right\|^2 + \frac{\eta^2\sum_{i=1}^{N}\sum_{\tau=1}^{t}p_{\tau,i}^2\sigma^2}{N^2K} \tag{47}$$

$$= \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\mathbb{E}\left\|\sum_{\tau=1}^{t}p_{\tau,i}\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\right\|^2 + \mathbb{E}\|\Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\|^2$$

$$- 2\sum_{\tau=1}^{t}p_{\tau,i}E\langle\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\rangle + \frac{\eta^2\sum_{i=1}^{N}\sum_{\tau=1}^{t}p_{\tau,i}^2\sigma^2}{N^2K} \tag{48}$$

$$\leq \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}(1 + B^2 + A^2(\sum_{\tau=1}^{t}p_{\tau,i}^2))\mathbb{E}\|\nabla f(\boldsymbol{\omega}_{t,i,k-1})\|^2 + (G^2 + D^2(\sum_{\tau=1}^{t}p_{\tau,i}^2))$$

$$+ \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\mathbb{E}\|\Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\|^2 - 2\sum_{\tau=1}^{t}p_{\tau,i}E\langle\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\rangle + \frac{\eta^2\sum_{i=1}^{N}\sum_{\tau=1}^{t}p_{\tau,i}^2\sigma^2}{N^2K} \tag{49}$$

$$\leq \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}2L^2(1 + B^2 + A^2(\sum_{\tau=1}^{t}p_{\tau,i}^2))\mathbb{E}\|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\|^2 + 2(1 + B^2 + A^2(\sum_{\tau=1}^{t}p_{\tau,i}^2))\|\nabla f(\boldsymbol{\omega}_t)\|^2 + (G^2 + D^2(\sum_{\tau=1}^{t}p_{\tau,i}^2))$$

$$+ \frac{\eta^2}{NK}\sum_{i=1}^{N}\sum_{k=1}^{K}\mathbb{E}\|\Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\|^2 - 2\sum_{\tau=1}^{t}p_{\tau,i}E\langle\nabla f_{t,i}(\boldsymbol{\omega}_{t,i,k-1}), \Delta_{t,i}(\boldsymbol{\omega}_{t,i,k-1})\rangle + \frac{\eta^2\sum_{i=1}^{N}\sum_{\tau=1}^{t}p_{\tau,i}^2\sigma^2}{N^2K}\,. \tag{50}$$

First combine (36) and A_2 in (38) to get equation (45). Use Lemma A.4 and Assumption 2 from equation (45) to derive inequality (46). Then use the definition in Lemma B.3 to get equation (47). Equation (48) is an extension of equation (47), and use Lemma 3.5 from equation (48) to inequality (49). Finally, use the fact that $\mathbb{E}\|\nabla f(\boldsymbol{\omega}_{t,i,k-1})\|^2 = \mathbb{E}\|\nabla f(\boldsymbol{\omega}_{t,i,k-1}) - \nabla f(\boldsymbol{\omega}_i) + \nabla f(\boldsymbol{\omega}_i)\|^2$, and use Lemma A.3 combined with the definition of L smooth, which is $\|\nabla f(\boldsymbol{\omega}_{t,i,k-1}) - \nabla f(\boldsymbol{\omega}_i)\|^2 \leq L^2\|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_i\|^2$, we get inequality (50).
Then according to equation (50), $\mathbb{E}\|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\|^2$ can be bounded as:

$$\mathbb{E}\|\boldsymbol{\omega}_{t,i,k} - \boldsymbol{\omega}_t\|^2 = \eta_l^2\mathbb{E}\left\|\sum_{\tau=1}^{k}g_{t,i}(\boldsymbol{\omega}_{t,i,\tau-1})\right\|^2 \tag{51}$$

$$\leq k\eta_l^2\sum_{\tau=1}^{k}\mathbb{E}\left\|\nabla\tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,\tau-1})\right\|^2 + k\eta_l^2\sigma^2\,. \tag{52}$$

Here we directly use the updating rule in algorithm 2 to get equation (51). Because $g_{t,i}(\boldsymbol{\omega}_{t,i,\tau-1}) = \nabla\tilde{f}_{t,i}(\boldsymbol{\omega}_{t,i,\tau-1}) + \boldsymbol{\nu}_{t,i,\tau-1}$, we use Lemma A.4 to get inequality (52).

Then the upper bound of $\mathbb{E}\left\|\nabla \tilde{f}_{t,i}(\boldsymbol{\omega})\right\|^2$ is derived below.

$$\mathbb{E}\left\|\nabla \tilde{f}_{t,i}(\boldsymbol{\omega})\right\|^2$$

$$= \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega}) - \Delta_{t,i}(\boldsymbol{\omega})\right\|^2 \tag{53}$$

$$= \mathbb{E}\left\|\sum_{\tau=1}^{t} p_{\tau,i}\nabla f_{\tau,i}(\boldsymbol{\omega})\right\|^2 - \frac{2}{\sum_{\tau=1}^{t} p_{\tau,i}}\sum_{\tau=1}^{t} p_{\tau,i}\langle \nabla f_{\tau,i}(\boldsymbol{\omega}), \Delta_{t,i}(\boldsymbol{\omega})\rangle + \mathbb{E}\|\Delta_{t,i}(\boldsymbol{\omega})\|^2 \tag{54}$$

$$\leq (1 + B^2 + A^2\sum_{\tau=1}^{t} p_{\tau,i}^2)\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 - \frac{2}{\sum_{\tau=1}^{t} p_{\tau,i}}\sum_{\tau=1}^{t} p_{\tau,i}\langle \nabla f_{\tau,i}(\boldsymbol{\omega}), \Delta_{t,i}(\boldsymbol{\omega})\rangle + \mathbb{E}\|\Delta_{t,i}(\boldsymbol{\omega})\|^2 + G^2 + D^2(\sum_{\tau=1}^{t} p_{\tau,i}^2)\,. \tag{55}$$

We use definition of $\Delta_{t,i}(\boldsymbol{\omega})$ in Lemma B.3 for equation (53), and use Lemma 3.5 to get inequality (55).
Here, in order to simplify the proof, we only concentrate on the condition that we won't lose information of previous rounds. That is, $\Delta_{t,i}(\boldsymbol{\omega}) = 0$ for any i, j, and $\boldsymbol{\omega}$. Then denote $c_{p_B,i} = 1 + B^2 + A^2\sum_{\tau=1}^{t} p_{\tau,i}^2$, $c_{p_G,i} = G^2 + D^2\sum_{\tau=1}^{t} p_{\tau,i}^2$, combine equation (55) and (52), and use the fact that $\mathbb{E}\|\nabla f(\boldsymbol{\omega})\|^2 \leq 2L(f(\boldsymbol{\omega}) - f(\boldsymbol{\omega}^*))$, we have

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{\omega}_{t,i,k} - \boldsymbol{\omega}_t\|^2\\ &\leq 2k^2\eta_l^2L^2c_{p_B,i}\mathbb{E}\|\boldsymbol{\omega}_{t,i,k-1} - \boldsymbol{\omega}_t\|^2 + 4k^2\eta_l^2Lc_{p_B,i}(f(\boldsymbol{\omega}_t) - f(\boldsymbol{\omega}^*)) + k^2\eta_l^2c_{p_G,i} + k\eta_l^2\sigma^2\,.\end{aligned} \tag{56}$$

That is,

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{\omega}_{t,i,k} - \boldsymbol{\omega}_t\|^2\\ &\leq \sum_{\tau=0}^{k-1}(4K^2\eta_l^2Lc_{p_B,i}(f(\boldsymbol{\omega}_t) - f(\boldsymbol{\omega}^*)) + K^2\eta_l^2c_{p_G,i} + K\eta_l^2\sigma^2)(2K^2\eta_l^2L^2c_{p_B,i})^\tau \qquad (57)\\ &\leq \frac{4K^2\eta_l^2Lc_{p_B,i}(f(\boldsymbol{\omega}_t) - f(\boldsymbol{\omega}^*)) + K^2\eta_l^2c_{p_G,i} + K\eta_l^2\sigma^2}{1 - 2K^2\eta_l^2L^2c_{p_B,i}}\,. \qquad (58)\end{aligned}$$

Define $c_{p_B} = \frac{1}{N}\sum_{i=1}^{N} c_{p_B,i}$ and $c_{p_B} = \frac{1}{N}\sum_{i=1}^{N} c_{p_G,i}$. Combine the inequalities (44), (50), and (58), we get

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{\omega}_t + \Delta\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2\\ &\leq (1 - \frac{\mu\eta}{2})\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 + (-2\eta + 4\eta^2Lc_{p_B} + \frac{8K^2\eta\eta_l^2L^2c_{p_B}(1 + \eta Lc_{p_B})}{1 - 2K^2\eta_l^2L^2c_{p_B}})(f(\boldsymbol{\omega}_t) - f^*)\\ &+ \eta^2c_{p_G} + \frac{\eta^2\sum_{\tau=1}^{t} p_\tau^2\sigma^2}{N^2K} + \frac{K\eta\eta_l^2L(Kc_{p_G} + \sigma^2)(1 + \eta Lc_{p_B})}{1 - 2K^2\eta_l^2L^2c_{p_B}}\,.\end{aligned} \tag{59}$$

Then we notice that

$$-2\eta + 4\eta^2Lc_{p_B} + \frac{8K^2\eta\eta_l^2L^2c_{p_B}(1 + \eta Lc_{p_B})}{1 - 2K^2\eta_l^2L^2c_{p_B}} \leq 0\,. \tag{60}$$

Based on the fact that $\eta = K\eta_l\eta_g$, above inequality is equivalent to

$$\eta \leq \frac{\sqrt{6\eta_g^2 + c_{p_B}\eta_g^4} - \sqrt{c_{p_B}\eta_g^4}}{6L\sqrt{c_{p_B}}}\,. \tag{61}$$

Then let $\eta \leq \frac{\sqrt{6\eta_g^2+c_{p_B}\eta_g^4}-\sqrt{c_{p_B}\eta_g^4}}{12L\sqrt{c_{p_B}}}$, we have $1 - 2K^2\eta_l^2L^2c_{p_B} \geq \frac{15-\eta_g^2c_{p_B}+\sqrt{6\eta_g^2c_{p_B}+\eta_g^4c_{p_B}^2}}{72}$, and then $\frac{1+\eta Lc_{p_B}}{1-2K^2\eta_l^2L^2c_{p_B}} \leq 12 - \frac{108}{15+\sqrt{6\eta_g^2c_{p_B}+\eta_g^4c_{p_B}^2}-\eta_g^2c_{p_B}} \leq 12$.

That is, assume $c_1 = c_{p_G} + \frac{\sum_{\tau=1}^{t} p_{\tau,i}^2 \sigma^2}{NK(\sum_{\tau=1}^{t} p_{t,i})^2}$, and $c_2 = \frac{12Lc_{p_G}}{\eta_g^2} + \frac{12L\sigma^2}{\eta_g^2 K}$, inequality (59) become

$$\mathbb{E}\|\boldsymbol{\omega}_t + \Delta\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 \leq (1 - \frac{\mu\eta}{2})\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2 - \eta(f(\boldsymbol{\omega}_t) - f^*) + c_1\eta^2 + c_2\eta^3 \,. \tag{62}$$

Notice that both c_1 and c_2 will change when t change. Thus, we rewrite c_1 at round t as $c_1(t)$, and c_2 at round t as $c_2(t)$, besides, rewrite c_{p_B}, c_{p_G} at round t as $c_{p_B}(t)$ and $c_{p_G}(t)$.
If local objective function f is $\mu-$strongly convex, then by choosing $\omega_F = \frac{1}{\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t\omega_t$, and $q_t = (1 - \frac{\mu\eta}{2})^{1-t}$, we have

$$f(\omega_F) - f^* \leq \frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t\left((1 - \frac{\mu\eta}{2})\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\| - \mathbb{E}\|\boldsymbol{\omega}_{t+1} - \boldsymbol{\omega}^*\| + c_1(t)\eta + c_2(t)\eta^2\right) \,. \tag{63}$$

Here we consider $\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_1(t)$ and $\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_2(t)$. When $p_{\tau,i} = \frac{1}{t}$ for all τ,

$$c_{p_G}(t) = \tilde{\mathcal{O}}\left(G^2 + \frac{D^2}{t}\right) \,. \tag{64}$$

Besides, by Lemma 3.5, when $\boldsymbol{\delta}_{t_1,i}$ and $\boldsymbol{\delta}_{t_2,i}$ are independent for all t_1 and t_2, we have tighter bound

$$c_{p_G}(t) = \tilde{\mathcal{O}}\left(\frac{G^2 + D^2}{t}\right) \,. \tag{65}$$

When we use FedAvg algorithm, $p_{\tau,i} = 0$ for $\tau = 1, 2, ..., t-1$, then

$$c_{p_G}(t) = \tilde{\mathcal{O}}\left(G^2 + D^2\right) \,, \tag{66}$$

By Lemma A.6, we have

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_1(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(G^2 + \frac{lnT}{T}\left(D^2 + \frac{\sigma^2}{NK}\right)\right) \,, \tag{67}$$

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_2(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(\frac{L(KG^2 + \sigma^2)}{\eta_g^2 K} + \frac{LD^2 lnT}{\eta_g^2 T}\right) \,. \tag{68}$$

Similarly, when $\boldsymbol{\delta}_{t_1,i}$ and $\boldsymbol{\delta}_{t_2,i}$ are independent for all t_1 and t_2, we have

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_1(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(\frac{lnT}{T}\left(G^2 + D^2 + \frac{\sigma^2}{NK}\right)\right) \,, \tag{69}$$

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_2(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(\frac{L\sigma^2}{\eta_g^2 K} + \frac{L(D^2 + G^2)lnT}{\eta_g^2 T}\right) \,. \tag{70}$$

When using FedAvg, above inequalities become

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_1(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(G^2 + D^2 + \frac{\sigma^2}{NK}\right) \,, \tag{71}$$

$$\frac{1}{\eta\sum_{t=1}^{T} q_t}\sum_{t=1}^{T} q_t c_2(t) \leq \frac{1}{\eta}\tilde{\mathcal{O}}\left(\frac{L\sigma^2}{\eta_g^2 K} + \frac{L(D^2 + G^2)}{\eta_g^2}\right) \,. \tag{72}$$

When $c_1(t)$ and $c_2(t)$ satisfy (67) and (68), by setting $c_{p_B} = (1 + A^2 + B^2)$ together with Lemma A.1, there is

$$f(\omega_F) - f^* \leq \frac{\mu\mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\mu\eta_{max}T) + c_1\eta + c_2\eta^2 \tag{73}$$

$$= \tilde{\mathcal{O}}\left(\frac{G^2}{\mu T} + \frac{D^2}{\mu T^2} + \frac{\sigma^2}{\mu NKT^2} + \frac{KG^2 + \sigma^2}{\mu^2\eta_g^2 KT^2} + \frac{D^2}{\mu^2\eta_g^2 T^3} + \frac{\mu\mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\frac{\mu T}{Lc_{p_B}})\right) \,. \tag{74}$$

When $c_1(t)$ and $c_2(t)$ satisfy (69) and (70), we have tighter bound

$$f(\omega_F) - f^* \leq \frac{\mu \mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\mu \eta_{max} T) + c_1 \eta + c_2 \eta^2 \tag{75}$$

$$= \tilde{\mathcal{O}}\left(\frac{G^2 + D^2}{\mu T^2} + \frac{\sigma^2}{\mu N K T^2} + \frac{\sigma^2}{\mu^2 \eta_g^2 K T^2} + \frac{G^2 + D^2}{\mu^2 \eta_g^2 T^3} + \frac{\mu \mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\frac{\mu T}{L c_{p_B}})\right). \tag{76}$$

When using FedAvg, $c_1(t)$ and $c_2(t)$ satisfy (71) and (72),

$$f(\omega_F) - f^* \leq \frac{\mu \mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\mu \eta_{max} T) + c_1 \eta + c_2 \eta^2 \tag{77}$$

$$= \tilde{\mathcal{O}}\left(\frac{G^2 + D^2}{\mu T} + \frac{\sigma^2}{\mu N K T} + \frac{\sigma^2}{\mu^2 \eta_g^2 K T^2} + \frac{G^2 + D^2}{\mu^2 \eta_g^2 T^2} + \frac{\mu \mathbb{E}\|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|^2}{2} exp(-\frac{\mu T}{L c_{p_B}})\right). \tag{78}$$

Besides, if the local objective function f is general convex, i.e., $\mu = 0$, we can directly use Lemma A.2. When $c_1(t)$ and $c_2(t)$ satisfy (67) and (68), denote $F = \|\boldsymbol{\omega}_0 - \boldsymbol{\omega}^*\|$

$$f(\omega_F) - f^* \leq \sum_{t=1}^{T} \frac{\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2}{\eta} - \frac{\mathbb{E}\|\boldsymbol{\omega}_{t+1} - \boldsymbol{\omega}^*\|^2}{\eta} + c_1 \eta + c_2 \eta^2 \tag{79}$$

$$= \tilde{\mathcal{O}}\left(\frac{c_{p_B} F}{T} + \frac{GF}{\sqrt{T}} + \frac{DF}{T} + \frac{\sigma F}{\sqrt{NKT}} + \sqrt[3]{\frac{(KG^2 + \sigma^2)F^4}{\eta_g^2 K T^2}} + \sqrt[3]{\frac{D^2 F^4}{\eta_g^2 T^3}}\right). \tag{80}$$

When $c_1(t)$ and $c_2(t)$ satisfy (69) and (70), we can derive a tighter bound below.

$$f(\omega_F) - f^* \leq \sum_{t=1}^{T} \frac{\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2}{\eta} - \frac{\mathbb{E}\|\boldsymbol{\omega}_{t+1} - \boldsymbol{\omega}^*\|^2}{\eta} + c_1 \eta + c_2 \eta^2 \tag{81}$$

$$= \tilde{\mathcal{O}}\left(\frac{c_{p_B} F}{T} + \frac{(G + D)F}{T} + \frac{\sigma F}{\sqrt{NKT}} + \sqrt[3]{\frac{\sigma^2 F^4}{\eta_g^2 K T^2}} + \sqrt[3]{\frac{(G^2 + D^2)F^4}{\eta_g^2 T^3}}\right). \tag{82}$$

For FedAvg, when $c_1(t)$ and $c_2(t)$ satisfy (71) and (72),

$$f(\omega_F) - f^* \leq \sum_{t=1}^{T} \frac{\mathbb{E}\|\boldsymbol{\omega}_t - \boldsymbol{\omega}^*\|^2}{\eta} - \frac{\mathbb{E}\|\boldsymbol{\omega}_{t+1} - \boldsymbol{\omega}^*\|^2}{\eta} + c_1 \eta + c_2 \eta^2 \tag{83}$$

$$= \tilde{\mathcal{O}}\left(\frac{c_{p_B} F}{T} + \frac{(G + D)F}{\sqrt{T}} + \frac{\sigma F}{\sqrt{NKT}} + \sqrt[3]{\frac{\sigma^2 F^4}{\eta_g^2 K T^2}} + \sqrt[3]{\frac{(G^2 + D^2)F^4}{\eta_g^2 T^2}}\right). \tag{84}$$

C. Experiment details

C.1. Noisy quadratic model

We use the noisy quadratic model introduced in (Zhang et al., 2019) to simulate the assumptions. Define $f(\boldsymbol{\omega}) = \boldsymbol{\omega}^T \mathbf{A} \boldsymbol{\omega} + \mathbf{B}^T \boldsymbol{\omega} + \mathbf{C}$, where $\boldsymbol{\omega} \in \mathbb{R}^n$ is the parameter to optimize, and $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{A} \succeq 0$, $\mathbf{B} \in \mathbb{R}^n$, and $\mathbf{C} \in \mathbb{R}$. We construct $\mathbf{A}$ by firstly construct a diagonal matrix Λ, and then construct $\mathbf{A}$ by let $\mathbf{A} = \mathbf{U}^T \Lambda \mathbf{U}$, where $\mathbf{U}$ is a unitary matrix. Here we control the eigenvalues of $\mathbf{A}$ to control the convexity of model, that is, $\mu \leq \lambda_{min}(\mathbf{A}) \leq \lambda_{max}(\mathbf{A}) \leq L$. Because Λ and $\mathbf{A}$ have same eigenvalues, it's simple to control the eigenvalues of $\mathbf{A}$ by control the eigenvalues of Λ.
To simulate Assumption 2, 4, and 3, we formulate the noisy gradient by

$$g_{t,i}(\boldsymbol{\omega}) = \mathbf{A}\boldsymbol{\omega} + \mathbf{B} + \boldsymbol{\delta}_i + \boldsymbol{\xi}_{t,i} + \boldsymbol{\nu}_{t,i,k}\,, \tag{85}$$

where $\boldsymbol{\delta}_i$ denotes the client drift of client j, $\boldsymbol{\xi}_{t,i}$ denotes the round drift of client j on round i, and $\boldsymbol{\nu}_{t,i,k}$ denotes the noise of gradient of client j on round i, iteration k. All of above drifts and noise are generated from Normal distribution with zero mean and different variance.

Method	SRD				BRD			
	SL-SC	LL-SC	SL-GC	LL-GC	SL-SC	LL-SC	SL-GC	LL-GC
FAVG	0.03	0.08	0.33	0.09	0.02	0.01	0.09	0.02
FPROX	0.04	0.07	0.35	0.09	0.02	0.01	0.26	0.02
CFL-R	0.2	0.09	0.35	0.09	0.25	0.08	0.3	0.08

Table 2: Best learning rate of different algorithms on noisy quadratic model

In practice, to show if the numerical results match our theoretical results, we tried eight different settings include large and small L, large and small round drift, and general and strongly convex conditions. For different types of L, we set $L = 20$ for large L, and $L = 5$ for small L. For convexity, we set $\mu = 1$ for strongly convex, and $\mu = 0$ for general convex. For different round drift, we set $\mathbb{E}\left\|\boldsymbol{\xi}_{t,i}\right\|^2 = 100$ for big round drift, and $\mathbb{E}\left\|\boldsymbol{\xi}_{t,i}\right\|^2 = 0.01$ for small round drift. Besides, because $\boldsymbol{\delta}_i$ and $\boldsymbol{\nu}_{t,i,k}$ are not important things for us to experiment, thus we use fixed variance for them, that is, $\mathbb{E}\|\boldsymbol{\delta}_i\|^2 = 0.01$, and $\mathbb{E}\|\boldsymbol{\nu}_{t,i,k}\|^2 = 0.00001$.
We do the gradient descent by let $\boldsymbol{\omega}_{t,i,k+1} = \boldsymbol{\omega}_{t,i,k} - \eta_l g_{t,i}(\boldsymbol{\omega}_{t,i,k})$. Notice that for convex functions, when it reaches global optimal point, we should have $\|A\boldsymbol{\omega} + B\| = 0$, and if it's far away from global optimal, $\|A\boldsymbol{\omega} + B\|$ will be large. Thus we use $\|A\boldsymbol{\omega} + B\|$ as loss value.
Besides, from the proof part, we know that CFL-R can tolerant larger learning rate. Table 2 shows the best learning rates for different FL algorithms on noisy quadratic model. In Table 2, we denote small round drift as SRD, big round drift as BRD, small L as SL, large L as LL, strongly convex as SC, and general convex as GC. Notice that the best learning rate of basic continual federated learning is the largest, and the difference between CFL-R and other FL baselines becomes large when models are strongly convex or with big round drift.

C.2. Algorithms

Algorithm 2 Exact CFL-R

Require: initial weights $\boldsymbol{\omega}_0$, global learning rate η_g, local learning rate η_l

for round t = 1,...,T **do**
  for client i = 1,...,N **do**
    $\mathbf{H}_{agg,i} \leftarrow \sum_{\tau=1}^{t-1} \mathbf{H}_{\tau,i}$.
    $\mathbf{G}_{agg,i} \leftarrow \sum_{\tau=1}^{t-1} \mathbf{G}_{\tau,i}(\boldsymbol{\omega}_{\tau,i,K}) - \mathbf{H}_{\tau,i}\boldsymbol{\omega}_{\tau,i,K}$.
  communicate $\boldsymbol{\omega}_t$, $\mathbf{H}_{agg,i}$, and $\mathbf{G}_{agg,i}$ to client j.
  for client i = 1,...,N **in parallel do**
    initialize local model $\boldsymbol{\omega}_{t,i,0} = \boldsymbol{\omega}_t$.
    for k = 1,...,K **do**
      compute mini-batch gradient $g_{t,i,k}(\boldsymbol{\omega}_{t,i,k-1})$.
      $\boldsymbol{\omega}_{t,i,k} \leftarrow \boldsymbol{\omega}_{t,i,k-1} - \frac{\eta_l}{t}(g_{t,i,k}(\boldsymbol{\omega}_{t,i,k-1}) + \mathbf{G}_{agg,i} + \mathbf{H}_{agg,i}\boldsymbol{\omega}_{t,i,k-1})$.
    $\mathbf{H}_{t,i}, \mathbf{G}_{t,i} \leftarrow calculateHessianAndGradient(\boldsymbol{\omega}_{t,i,K})$.
    communicate $\Delta\boldsymbol{\omega}_{t,i} \leftarrow \boldsymbol{\omega}_{t,i,K} - \boldsymbol{\omega}_t$, $\mathbf{H}_{t,i}$, $\mathbf{G}_{t,i}$.
  $\Delta\boldsymbol{\omega}_t \leftarrow \frac{\eta_g}{N}\sum_{i=1}^{N} \Delta\boldsymbol{\omega}_{t,i}$.
  $\boldsymbol{\omega}_{t+1} \leftarrow \boldsymbol{\omega}_t + \Delta\boldsymbol{\omega}_t$.

D. Related Work

Federated Learning on heterogeneous environments. The de facto standard algorithm is FedAvg (McMahan et al., 2017; Lin et al., 2020), where multiple local SGD steps are performed on the available clients to alleviate the communication bottleneck. While communication efficient, the heterogeneity, like system heterogeneity (Li et al., 2018; Wang et al., 2020; Mitra et al., 2021; Diao et al., 2021) and statistical/objective heterogeneity (Li et al., 2018; Wang et al., 2020; Mitra et al., 2021; Karimireddy et al., 2020b;a), causes inconsistent optimization objective and drifted clients model, and thus significantly hinders the federated optimization.
A line of work has been proposed to address the heterogeneity in FL. FedProx (Li et al., 2018) adds the regularization

term on the distance of local and global models when performing local training—similar formulations can be seen in other recent FL works (Hanzely & Richtárik, 2020; Dinh et al., 2020; Li et al., 2020a) for various purposes. To alleviate the issue of objective heterogeneity e.g. caused by heterogeneous data, works like SCAFFOLD (Karimireddy et al., 2020b;a; Mitra et al., 2021) introduce the idea of variance reduction on the client local update steps. FedNova (Wang et al., 2020) further proposes a general framework to unify FedAvg and FedProx, and argue to normalize the local updates when averaging, to eliminate the heterogeneity caused by different number of local update steps. However, most of these prior works focus on the fixed heterogeneity across clients, and over the whole optimization procedure. In this work, we consider the novel time-evolving data heterogeneity.

The theoretical analysis on the convergence of FedAvg can date back to the parallel SGD analysis on the identical functions (Zinkevich et al., 2010) and recently is refined by (Stich, 2019; Stich & Karimireddy, 2020; Patel & Dieuleveut, 2019; Khaled et al., 2020; Woodworth et al., 2020b). For the analysis on heterogeneous data, Li et al. (2020c) first give the convergence rate of FedAvg on non-iid datasets with random selection, and assume that the client optimum are ϵ-close. Woodworth et al. (2020a); Khaled et al. (2020) give tighter convergence rates under the assumption of bounded gradient drift. All above works give a $\mathcal{O}\left(\frac{1}{T}\right)$ convergence rate for convex local objective functions. More recently, Karimireddy et al. (2020b); Koloskova et al. (2020) give the convergence analysis of local SGD for non-convex objective functions under bounded gradient noise assumptions, and get a $\mathcal{O}\left(\frac{1}{\sqrt{T}}\right)$ convergence rate. Yang et al. (2021) prove a tighter convergence rate and show that the maximum number of local steps can be improved to $\frac{T}{m}$ in full worker participation, compared with $\frac{T^{\frac{1}{3}}}{m}$ in Yu et al. (2019); Karimireddy et al. (2020b). Our theoretical analysis framework covers more challenging time-varying data heterogeneity in FL, which has not been considered in the community yet—our rate can be simplified to the standard FL scenario, matching the tight analysis in Karimireddy et al. (2020b).

Continual Learning. Continual learning, a.k.a. incremental learning or lifelong learning, aims to learn from (time-varying) sequential data while avoiding the issue of *catastrophic forgetting* (French, 1999; Kirkpatrick et al., 2017). Though there exists a large amount of works, from the perspectives of regularization (Kirkpatrick et al., 2017; Li & Hoiem, 2017; Zenke et al., 2017), experience replay (Castro et al., 2018; Rebuffi et al., 2017), and dynamic architectures (Maltoni & Lomonaco, 2019; Rusu et al., 2016), the efforts on the theoretical understanding is limited. Only a very recent preprint (Yin et al., 2020b) provides a viewpoint of regularization-based continual learning by formulating it as a second-order Taylor approximation of the loss function of each task, which can derive many existing algorithms. However, we consider collaborative learning and most of the prior work only consider the single worker case. Besides, we derive a more precise convergence rate for CL using mini-batch SGD.

Continual Federated Learning. To the best of our knowledge, the scenario of CFL was first appeared in Bui et al. (2018) for federated training Bayesian Neural Network and continual learning for Gaussian Process models—it is orthogonal to our optimization aspect in this paper. In a very recent deep learning paper, FedWeIT (Yoon et al., 2020) extends the regularization-based method to enable learning a sequence of client tasks; however, their approach has no theoretical guarantee.