FedBoost: Communication-Efficient Algorithms for Federated Learning

Jenny Hamer [1] **Mehryar Mohri** [1 2] **Ananda Theertha Suresh** [1]

Abstract

Communication cost is often a bottleneck in federated learning and other client-based distributed learning scenarios. To overcome this, several gradient compression and model compression algorithms have been proposed. In this work, we propose an alternative approach whereby an ensemble of pre-trained base predictors is trained via federated learning. This method allows for training a model which may otherwise surpass the communication bandwidth and storage capacity of the clients to be learned with on-device data through federated learning. Motivated by language modeling, we prove the optimality of ensemble methods for density estimation for standard empirical risk minimization and agnostic risk minimization. We provide communication-efficient ensemble algorithms for federated learning, where per-round communication cost is independent of the size of the ensemble. Furthermore, unlike previous work on gradient compression, our algorithm helps reduce the cost of both server-to-client and client-to-server communication.

1. Introduction

With the growing prevalence of mobile phones, sensors, and other edge devices, designing communication-efficient techniques for learning using client data is an increasingly important area in distributed machine learning. *Federated learning* is a setting where a centralized model is trained on data that remains distributed among the clients (Konečný et al., 2016b; McMahan et al., 2017; Mohri et al., 2019). Since the raw local data are not sent to the central server coordinating the training, federated learning does not directly expose user data to the server and can be combined with cryptographic techniques for additional layers of privacy.

[1]Google Research, New York, NY, USA [2]Courant Institute of Mathematical Sciences, New York, NY, USA. Correspondence to: Jenny Hamer <hamer@google.com>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

Federated learning has been shown to perform well on several tasks, including next word prediction (Hard et al., 2018; Yang et al., 2018), emoji prediction (Ramaswamy et al., 2019), decoder models (Chen et al., 2019b), vocabulary estimation (Chen et al., 2019a), low latency vehicle-to-vehicle communication (Samarakoon et al., 2018), and predictive models in health (Brisimi et al., 2018).

Broadly, in federated learning, at each round, the server selects a subset of clients who receive the current model. These clients then run a few steps of *stochastic gradient descent* locally and send back the model updates to the server. Training is repeated until convergence. Given the distributed nature of clients, federated learning raises several research challenges, including privacy, optimization, systems, networking, and communication bottleneck problems. Of these, communication bottleneck has been studied extensively in terms of compression of model updates from client to servers (Konečný et al., 2016b;a; Suresh et al., 2017).

This line of work requires that the model size be small enough to fit into the client devices' memory. This assumption holds in several applications. For example, the size of typical state-of-the-art server-side language models is in the order of several hundreds of megabytes (Kumar et al., 2017). Similarly, that of speech recognition models, which admit a few million parameters, is on the order of hundreds of megabytes (Sak et al., 2015). However, other applications motivate the need for larger models and alternative solutions. Such larger models can be used, for example, in server-side inference. They can also be further processed either by distillation techniques (Hinton et al., 2015), or compressed using quantization (Wu et al., 2016) or pruning (Han et al., 2015) for on-device inference. This prompts the following question: Can we learn very large models in federated learning that may not fit in client devices' memory?

We present a solution to this problem by showing that large models can be learned via federated learning using *ensemble methods*. Ensemble methods are general techniques in machine learning for combining several *base predictors* or experts to create a single more accurate model. In the standard supervised learning setting, they include prominent techniques such as bagging, boosting, stacking, error-

correction techniques, Bayesian averaging, or other averaging schemes (Breiman, 1996; Freund et al., 1999; Smyth & Wolpert, 1999; MacKay, 1991; Freund et al., 2004). Ensemble methods often significantly improve performance in practice in a variety of tasks (Dietterich, 2000), including image classification (Kumar et al., 2016) and language models (Jozefowicz et al., 2016).

One of the main bottlenecks in federated learning is communication efficiency, which is determined by the number of parameters sent from the server to the clients and from clients to the server at each round (Konečný et al., 2016b). Since the current iterate of the model is sent to all participating clients during each round, directly applying known ensemble methods to federated learning could cause a significant or even infeasible blow-up in communication costs due to transmitting every predictor, every round.

We propose FEDBOOST, a new communication-efficient ensemble method that is theoretically motivated and has significantly smaller communication overhead, compared to the existing algorithms. In addition to the communication-efficiency, ensemble methods offer several advantages in federated learning. They include computational speedups, convergence guarantees, privacy, and the optimality of the solution for density estimation for which language modeling is a special case. We list several of their other advantages in this context below:

- Pre-trained base predictors: base predictors can be pre-trained on publicly available data, thus reducing the need for user data in training.
- Convergence guarantee: ensemble methods often require training relatively few parameters, which typically results in far fewer rounds of optimization and faster convergence compared to training the entire model from scratch.
- Adaptation or drifting over time: user data may change over time, but, in the ensemble approach, we can keep the base predictors fixed and retrain the ensemble weights whenever the data changes (Mohri & Medina, 2012).
- Differential privacy (DP): federated learning can be combined with global DP to provide an additional layer of privacy (Kairouz et al., 2019). Training only the ensemble weights via federated learning is well-suited for DP since the utility-privacy trade-off depends on the number of parameters being trained (Bassily et al., 2014). Furthermore, this learning problem is typically a convex optimization problem for which DP convex optimization can give better privacy guarantees.

1.1. Related work

The problem of learning ensembles is closely related to that of *multiple source domain adaptation* (MSDA), first formalized and analyzed theoretically by Mansour, Mohri, and Rostamizadeh (2009b;a) and Hoffman et al. (2018) and later studied for various applications such as object recognition (Hoffman et al., 2012; Gong et al., 2013a;b). Recently, Zhang et al. (2015) studied a causal formulation of this problem for a classification scenario, using the same combination rules as in (Mansour et al., 2009b;a; Hoffman et al., 2018).

There are several key differences between MSDA and our approach. For MSDA, (Mansour et al., 2009b;a; Hoffman et al., 2018) showed that the optimal combination of single source models are not ensembles and one needs to consider feature weighted ensembles. However, the focus of their approach was regression and classification under covariate shift, where the labeling function is assumed to be invariant or approximately invariant across domains. In contrast, our paper focuses on density estimation and we show that for density estimation ensemble methods are optimal.

Communication-efficient algorithms for distributed optimization has been the focus of several studies, both in federated learning (Konečný et al., 2016b;a; Suresh et al., 2017; Caldas et al., 2018) and in other distributed settings (Stich et al., 2018; Karimireddy et al., 2019; Basu et al., 2019). However, much of this previous work focuses on gradient compression and thus is only applicable to client-to-server communication, and does not apply to server-to-client communication. Recently, Caldas et al. (2018) proposed algorithms for reducing server to client communication and evaluated them empirically.

In contrast, the client-to-server communication is negligible in ensemble methods when only the mixing weights are learned via federated learning, since that accounts for very few parameters. The main focus of this work is addressing the server-to-client communication bottleneck. We propose communication-efficient methods for ensembles and provide convergence guarantees.

2. Learning scenario

In this section, we introduce the main problem of learning *federated ensembles*. We first outline the general problem, then discuss two important learning scenarios: *standard federated learning*, which assumes the union of samples from all domains is distributed uniformly, and *agnostic federated learning* where the test distribution is an unknown mixture of the domains.

We begin by introducing some general notation and definitions used throughout this work. We denote by $\mathcal{X}$ the

input space and $\mathcal{Y}$ the output space, with data samples $(x, y) \in \mathcal{X} \times \mathcal{Y}$. Consider the multi-class classification problem where $\mathcal{Y}$ represents a finite set of classes, and $\mathcal{H}$ a set of hypotheses where $h \in \mathcal{H}$ is a map of the form $h\colon \mathcal{X} \to \Delta_{\mathcal{Y}}$, where $\Delta_{\mathcal{Y}}$ is the probability simplex over $\mathcal{Y}$.

Denote by ℓ a loss function over $\Delta_{\mathcal{Y}} \times \mathcal{Y}$, where the loss for a hypothesis $h \in \mathcal{H}$ over a sample $(x, y) \in \mathcal{X} \times \mathcal{Y}$ is given by $\ell(h(x), y)$. For example, one common loss used in statistical parameter estimation is the squared error loss, given by $\mathbb{E}_{y' \sim h(x)}[\|y' - y\|]_2^2$. For a general loss ℓ, the expected loss of h with respect to a distribution $\mathcal{D}$ over $\mathcal{X} \times \mathcal{Y}$ is denoted by $\mathcal{L}_{\mathcal{D}}(h)$:

$$\mathcal{L}_{\mathcal{D}}(h) = \mathop{\mathbb{E}}_{(x,y)\sim\mathcal{D}}[\ell(h(x), y)].$$

Motivated by language modeling efforts in federated learning (Hard et al., 2018), a particular sub-problem of interest is density estimation, which is a special case of classification with $\mathcal{X} = \emptyset$ and $\mathcal{Y}$ is the set of domain elements.

2.1. Losses in federated learning

Following (Mohri et al., 2019), let the clients belong to one of p domains $\mathcal{D}_1, \dots \mathcal{D}_p$. While the distributions $\mathcal{D}_k$ may coincide with the clients, there is more flexibility in considering domains representing clusters of clients, particularly when the data are partitioned over a very large number of clients. In practice, the distributions $\mathcal{D}_k$ are not accessible and instead we observe samples $S_1, \dots, S_p$, drawn from domains with distribution $\mathcal{D}_k$ where each sample $S_k = ((x_{k,1}, y_{k,1}), \dots, (x_{k,m_k}, y_{k,m_k}) \in (\mathcal{X} \times \mathcal{Y})^{m_k}$ is of size m_k. Let $\hat{\mathcal{D}}_k$ denote the empirical distribution of $\mathcal{D}_k$. The empirical loss of an estimator h for domain k is

$$\mathcal{L}_{\hat{\mathcal{D}}_k}(h) = \frac{1}{m_k} \sum_{(x,y)\in\hat{\mathcal{D}}_k} \ell(h(x), y). \tag{1}$$

In standard federated learning, the central server minimizes the loss over the uniform distribution of all samples, $\overline{\mathcal{U}} = \sum_{p=1}^{k} \frac{m_k}{m} \hat{\mathcal{D}}_k$, with the assumed target distribution given by $\overline{\mathcal{U}} = \sum_{k=1}^{p} \frac{m_k}{m} \mathcal{D}_k$. This optimization problem is defined as

$$\min_{h\in\mathcal{H}} \mathcal{L}_{\overline{\mathcal{U}}}(h), \quad \text{where} \quad \mathcal{L}_{\overline{\mathcal{U}}}(h) = \sum_{k=1}^{p} \frac{m_k}{m} \mathcal{L}_{\hat{\mathcal{D}}_k}(h). \tag{2}$$

In federated learning, the target distribution may be significantly different from $\overline{\mathcal{U}}$ and hence (Mohri et al., 2019) proposed *agnostic federated learning*, which accounts for heterogeneous data distribution across clients. Let Δ_p denote the probability simplex over the p domains. For a $\lambda \in \Delta_p$, let $\mathcal{D}_\lambda = \sum_{k=1}^{p} \lambda_k \mathcal{D}_k$ be the mixture of distributions with unknown mixture weight λ. The learner's goal is to determine a solution which performs well for any $\lambda \in \Delta_p$ or any convex subset $\Lambda \subseteq \Delta_p$. More concretely, the objective is to find the hypothesis $h \in \mathcal{H}$ that minimizes the *agnostic loss*, given by

$$\mathcal{L}_{\mathcal{D}_\Lambda}(h) = \max_{\lambda\in\Lambda} \mathcal{L}_{\mathcal{D}_\lambda}(h). \tag{3}$$

In this paper, we present algorithms and generalization bounds for ensemble methods for both of the above losses.

2.2. Federated ensembles

We assume that we have a collection $\mathbb{H}$ of q pre-trained hypotheses $\mathbb{H} = (h_1, ..., h_q)$ (predictors or estimators depending on the task). It is desirable that there exists one good predictor for each domain k, though in principle these predictors can be trained in any way, on user data or public data. The goal is to learn a corresponding set of weights $\alpha = \{\alpha_1, ..., \alpha_q\}$ to construct an ensemble $\sum_{k=1}^{q} \alpha_k h_k$ that minimizes the standard or agnostic loss. Furthermore, since we focus mainly on density estimation, we assume that $\sum_k \alpha_k = 1$. As stated in Section 1.1, unlike MSDA, we show that such ensembles are optimal for density estimation. Thus the set of hypothesis we consider are

$$\mathcal{H} = \left\{ \sum_{k=1}^{q} \alpha_k h_k : \alpha_k \geq 0, \forall k, \sum_{k=1}^{q} \alpha_k = 1 \right\}.$$

With this set of hypotheses, we minimize both the standard loss (2) and the agnostic loss (3). In the next section, we present theoretical guarantees for this learning scenario, and algorithms to learn federated ensembles in Section 4.

3. Optimality of ensemble methods for density estimation

Density estimation is a fundamental learning problem with wide applications including language modeling, where the goal is to assign probability distribution over sentences. In this section, we show that for a general class of divergences called *Bregman divergences*, ensemble methods are optimal for both standard and agnostic federated learning, for which we then provide generalization bounds.

3.1. Definitions

Recall that density estimation is a special case of classification where $\mathcal{X} = \emptyset$ and $\mathcal{Y}$ is the set of domain elements. An estimator h assigns probability to all elements of $\mathcal{Y}$. For notational simplicity, let h_y denote the probability assigned by h to an element $y \in \mathcal{Y}$. For example, in language modeling $\mathcal{Y}$ is the set of all sequences and an estimator, often a recurrent neural network, assigns probabilities to the set of all sequences. To measure distance between distributions, we use the Bregman divergence, which is defined as follows.

Definition 1 ((Bregman, 1967)). *Let $F\colon \mathcal{C} \to \mathbb{R}$ be a convex and differentiable function defined on a non-empty convex open set $\mathcal{C} \subseteq \Delta_{\mathcal{Y}}$.*[1] *Then, the* Bregman divergence *between a distribution $\mathcal{D} \in \mathcal{C}$ and an estimator $h \in \mathcal{C}$ is defined by*

$$\mathsf{B}_F(\mathcal{D} \parallel h) = F(\mathcal{D}) - F(h) - \langle \nabla F(h), \mathcal{D} - h \rangle,$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product in $\Delta_{\mathcal{Y}}$.

Throughout this section, we choose $\mathcal{C} = \Delta_{\mathcal{Y}+}$, given by

$$\Delta_{\mathcal{Y}+} = \{p : \sum_{y \in \mathcal{Y}} p(y) = 1, p(y) > 0, \forall y \in \mathcal{Y}\},$$

and restrict all distributions and hypotheses to be in $\Delta_{\mathcal{Y}+}$.

The standard squared loss is a Bregman divergence with function $F\colon x \mapsto \|x\|_2^2$ defined over $\mathbb{R}^d$, where d is the dimension. Similarly, the *generalized Kullback-Leibler (KL) divergence* or *unnormalized relative entropy* is a Bregman divergence defined by $F\colon x \mapsto \sum_{i=1}^d x_i \log x_i - x_i$ defined over $\mathbb{R}_+^d = \{x : x_i > 0, \forall i \leq d\}$. We note that Bregman divergences are non-negative and in general asymmetric.

For a domain $\mathcal{D}_k$ and a hypothesis h, the loss is thus

$$\mathcal{L}_{\mathcal{D}_k}(h) = \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big).$$

For a mixture of domains $\mathcal{D}_\lambda$ and a hypothesis h, we define the loss as

$$\mathcal{L}_{\mathcal{D}_\lambda}(h) = \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right).$$

3.2. Optimality for density estimation

We first show that for any Bregman divergence, given a sufficiently large hypothesis set $\mathcal{H}$, that the minimizer of (2) and (3) is a linear combination of the distributions $\mathcal{D}_k$. Thus, if we have access to infinitely many samples from the true distributions $\mathcal{D}_k$, then we can find the best hypothesis for each distribution $\mathcal{D}_k$ and then use their ensemble to obtain optimal estimators for both standard and agnostic losses. We first show the result for the standard loss. The result is similar to (Banerjee et al., 2005, Lemma 1); we provide the proof in Appendix A.1 for completeness.

Lemma 1. *Let the loss be a Bregman divergence B_F. Then, for any $\lambda \in \Lambda \subseteq \Delta_p$, if $h^* = \sum_{k=1}^p \lambda_k \mathcal{D}_k$ is in $\mathcal{H}$, then it is a minimizer of $h \mapsto \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$. If F is further strictly convex, then it is the unique minimizer.*

Using this lemma, we show that ensembles are also optimal for the agnostic loss. Due to space constraints, the proof if relegated to Appendix A.2.

[1] Some of our results require strict convexity which we highlight when necessary.

Lemma 2. *Let the loss be a Bregman divergence B_F with F strictly convex and assume that $\operatorname{conv}(\{\mathcal{D}_1, ..., \mathcal{D}_p\}) \subseteq \mathcal{H}$. Observe that B_F is jointly convex in both arguments. Then, for any convex set $\Lambda \subseteq \Delta_p$, the solution of the optimization problem $\min_{h \in \mathcal{H}} \max_{\lambda \in \Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ exists and is in $\operatorname{conv}(\{\mathcal{D}_1, ..., \mathcal{D}_p\})$.*

3.3. Ensemble bounds

The results just presented assume that $\sum_{k=1}^p \lambda_k \mathcal{D}_k$ is in $\mathcal{H}$. In practice, however, for each $k \in [p]$, we only have access to an estimate of $\mathcal{D}_k$, $h_k \in \mathcal{H}$. In this section, we will assume that, for each $k \in [p]$, h_k is a reasonably good estimate of the distribution $\mathcal{D}_k$ in the following sense:

$$\forall k, \exists h \in \mathbb{H} \text{ such that } \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) \leq \epsilon, \tag{4}$$

and analyze how well the ensemble output performs on the true mixture. Let h_α denote the ensemble $\sum_{\ell=1}^q \alpha_\ell h_\ell$.

Lemma 3. *Assume that (4) holds and that the Bregman divergence is jointly convex. Then, the following inequality holds:*

$$\min_\alpha \sum_{k=1}^p \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big) \leq \sum_{k=1}^p \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^p \lambda_\ell \mathcal{D}_\ell\Big) + \epsilon.$$

Proof. By (4), for every distribution $\mathcal{D}_k$, there exists a $h \in \mathbb{H}$ such that $\mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) \leq \epsilon$. Let h_k be one such hypothesis. Hence,

$$\min_\alpha \sum_{k=1}^p \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big) \leq \sum_{k=1}^p \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^p \lambda_\ell h_\ell\Big).$$

By (12) (Appendix A.1),

$$\begin{aligned} &\sum_{k=1}^p \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^p \lambda_\ell h_\ell\Big) \\ &= \mathsf{B}_F\Big(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel \sum_{\ell=1}^p \lambda_\ell h_\ell\Big) \\ &\quad + \sum_{k=1}^p \lambda_k F(\mathcal{D}_k) - F\Big(\sum_{k=1}^p \lambda_k \mathcal{D}_k\Big). \end{aligned} \tag{5}$$

By the joint convexity of the Bregman divergence, the following holds:

$$\begin{aligned} \mathsf{B}_F\Big(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel \sum_{\ell=1}^p \lambda_\ell h_\ell\Big) &\leq \sum_{k=1}^p \lambda_k \mathsf{B}_F(\mathcal{D}_k \parallel h_k) \\ &\leq \sum_{k=1}^p \lambda_k \epsilon = \epsilon. \end{aligned}$$

Next, observe that

$$\sum_{k=1}^{p} \lambda_k F(\mathcal{D}_k) - F\Big(\sum_{k=1}^{p} \lambda_k \mathcal{D}_k\Big) = \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell \mathcal{D}_\ell\Big). \quad (6)$$

This completes the proof. □

We now show a similar result for the agnostic loss. Our analysis makes use of the *information radius* R, an information-theoretic quantity defined as follows:

$$R = \max_{\lambda} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell \mathcal{D}_\ell\Big).$$

Theorem 1. *Assume that (4) holds and that the Bregman divergence is jointly convex in both arguments. Then, the following inequality holds:*

$$\min_{\alpha} \max_{\lambda} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big) \leq R + \epsilon.$$

Proof. The function $f : \Delta_p \times \Delta_q \to \mathbb{R}$ given by $f(\lambda, \alpha) = \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h_\alpha\big)$ is well-defined since all distributions $\mathcal{D}_k$ and hypotheses h_k lie in $\Delta_{\mathcal{Y}_+}$. Furthermore, f is linear (and hence concave) in λ. By the standard convexity of Bregman divergence, f is convex in α. The sets Δ_p and Δ_q are compact and convex by definition. Hence, by Sion's minimax theorem,

$$\min_{\alpha} \max_{\lambda} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big) = \max_{\lambda} \min_{\alpha} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big).$$

By (4), for every distribution $\mathcal{D}_k$, there exists a $h \in \mathbb{H}$ such that $\mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) \leq \epsilon$. Let h_k be one such hypothesis. Therefore,

$$= \max_{\lambda} \min_{\alpha} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_\alpha\Big) \leq \max_{\lambda} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell h_\ell\Big).$$

By (5) and (6), we can write:

$$\begin{aligned}
&\leq \max_{\lambda} \sum_{k=1}^{p} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell h_\ell\Big) \\
&\leq \max_{\lambda} \mathsf{B}_F\Big(\sum_{k} \lambda_k \mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell h_\ell\Big) \\
&+ \max_{\lambda} \sum_{k} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell \mathcal{D}_\ell\Big) \\
&= \max_{\lambda} \mathsf{B}_F\Big(\sum_{k} \lambda_k \mathcal{D}_k \parallel \sum_{\ell=1}^{p} \lambda_\ell h_\ell\Big) + R \\
&\leq \max_{\lambda} \sum_{k} \lambda_k \mathsf{B}_F\Big(\mathcal{D}_k \parallel h_k\Big) + R \\
&\leq \epsilon + R,
\end{aligned}$$

where the penultimate inequality follows from the joint convexity of the Bregman divergence. □

4. Algorithms

We propose algorithms for learning ensembles in the standard and agnostic federated learning settings which addresses the *server-to-client* communication bottleneck. Suppose we have a set of pre-trained base predictors or hypotheses, which we denote by $\mathbb{H} \triangleq \{h_1, ..., h_q\}$. In standard ensemble methods, the full set of hypotheses would be sent to each participating client. In practice, however, this may be infeasible due to limitations in communication bandwidth between the server and clients, as well as in memory and computational capacity of the clients.

To overcome this, we suggest a sampling method which sends a fraction of the hypotheses to the clients. While this reduces the communication complexity, it also renders the overall gradients biased, and the precise characterization of the ensemble convergence is not clear.

Recall that the optimization problem is over the ensemble weights $\alpha \in \Delta_q$, since the base estimators h_k are fixed. We rewrite the losses in terms of α and use the following notation. Let $\mathsf{L}_k(\alpha)$ denote the empirical loss, $\mathcal{L}_{\hat{\mathcal{D}}_k}(h_\alpha)$, of the ensemble on domain k over m_k samples:

$$\mathsf{L}_k(\alpha) = \frac{1}{m_k} \sum_{i=1}^{m_k} \ell(h_\alpha(x_{k,i}), y_{k,i}) \quad (7)$$

where h_α denotes the ensemble weighted by mixture weight α,

$$h_\alpha = \sum_{k=1}^{q} \alpha_k h_k.$$

Let C be the maximum number of base predictors that we can send to the client at each round, which denotes the com-

munication efficiency. In practice we prefer $C \ll q$ (particularly when q is large) and the communication cost per round would be independent of the size of the ensemble.

4.1. Standard federated ensemble

As in Section 2.2, the objective is to learn the coefficients $\alpha \in \Delta_q$ for an ensemble of the pre-trained base estimators h_k. In the new notation, this can be written as

$$\min_{\alpha \in \Delta_q} \mathsf{L}_{\overline{\mathcal{U}}}(\alpha), \quad \text{where } \mathsf{L}_{\overline{\mathcal{U}}}(\alpha) = \sum_{k=1}^{p} \frac{m_k}{m} \mathsf{L}_k(\alpha).$$

For the above minimization, we introduce a variant of the mirror descent algorithm (Nemirovski & Yudin, 1983), a generalization of gradient descent algorithm. Since a naive application of mirror descent would use the entire collection of hypotheses for the ensemble, we propose FEDBOOST, a communication-efficient federated ensemble algorithm, given in Figure 1.

During each round t of training, FEDBOOST samples two subsets at the server: a subset of pre-trained hypotheses, where each is selected with probability $\gamma_{k,t}$, denoted by $\mathbb{H}_t$, and a random subset of N clients, denoted by S_t. We define the following Bernoulli indicator by

$$\mathbf{1}_{k,t} \triangleq \begin{cases} 1 & \text{if } h_k \in \mathbb{H}_t, \\ 0 & \text{if } h_k \notin \mathbb{H}_t. \end{cases}$$

Under this random sampling, the ensemble at time t is $\sum_{k=1}^{q} \alpha_{k,t} h_k \mathbf{1}_{k,t}$. Observe that since

$$\mathbb{E}\left[\sum_{k=1}^{q} \alpha_{k,t} h_k \mathbf{1}_{k,t}\right] = \sum_{k=1}^{q} \alpha_{k,t} h_k \gamma_{k,t},$$

this is a biased estimator of the ensemble $\sum_{k=1}^{q} \alpha_k h_k$; we correct this by dividing by $\gamma_{k,t}$ to give the unbiased estimate of the ensemble:

$$\mathbb{E}\left[\sum_{k=1}^{q} \frac{\alpha_{k,t} h_k \mathbf{1}_{k,t}}{\gamma_{k,t}}\right] = \sum_{k=1}^{q} \alpha_{k,t} h_k.$$

We provide and analyze two ways of selecting $\gamma_{k,t}$ based on the communication-budget C: *uniform sampling* and *weighted random sampling*. Under uniform sampling, $\gamma_{k,t} = \frac{C}{q}$. Using weighted random sampling, $\gamma_{k,t}$ is proportional to the relative weight of h_k:

$$\gamma_{k,t} \triangleq \begin{cases} 1 & \text{if } \alpha_{k,t} C > 1 \\ \alpha_{k,t} C & \text{otherwise.} \end{cases} \tag{8}$$

We now provide convergence guarantee for FEDBOOST. To this end, we make the following set of assumptions.

Algorithm FEDBOOST

Initialization: pre-trained $\mathbb{H} = \{h_1, ..., h_q\}$, $\alpha_1 = \text{argmin}_{x \in \Delta_q} F(x)$, $\gamma_{k,1} = [\frac{1}{q}, ..., \frac{1}{q}]$.
Parameters: rounds $T \in \mathbb{Z}^+$, step size $\eta > 0$.
For $t = 1$ to T:

1. Uniformly sample N clients: S_t
2. Obtain $\mathbb{H}_t$ via uniform sampling or (8).
3. For each client j:
   (a) Send current ensemble model $\sum_{k \in \mathbb{H}_t} \tilde{\alpha}_{k,t} h_k$ to client j, where $\tilde{\alpha}_t = \frac{\alpha_{k,t}}{\gamma_{k,t}}$ if $h_k \in \mathbb{H}_t$, else 0.
   (b) Obtain the gradient update $\nabla \mathsf{L}_j(\tilde{\alpha}_t)$ and send to server.
4. $\delta_t \mathsf{L} = \sum_{j \in S_t} \frac{m_j}{m} \nabla \mathsf{L}_j(\tilde{\alpha}_t)$, where $m = \sum_{j \in S_t} m_j$
5. $v_{t+1} = [\nabla F]^{-1}(\nabla F(\alpha_t) - \eta \delta_t \mathsf{L})$, $\alpha_{t+1} = \mathrm{BP}(v_{t+1})$

Output: $\alpha^A = \frac{1}{T} \sum_{t=1}^{T} \alpha_t$

Subroutine BP (Bregman projection)

Input: x', Δ_q **Output:** $\text{argmin}_{x \in \Delta_q} \mathsf{B}_F(x \parallel x')$

Figure 1. Pseudocode of the FEDBOOST algorithm.

Properties 1. *Assume the following properties about the function F, the Bregman divergence B_F and the loss function L:*

1. *F is strongly convex with parameter $\sigma > 0$.*
2. *$\alpha_* \triangleq \max_{\alpha \in \Delta_q} \|\alpha\|$.*
3. *For any two α and α', $\mathsf{B}_F(\alpha \parallel \alpha') \leq r_\alpha$.*
4. *The dual norm of the third derivative tensor product is bounded:*
   $\max_{\|w\|_2 \leq 1} \|\nabla^3 \ell(h(x), y) \otimes w \otimes w\|_* \leq M$.
5. *The norm of the gradient is bounded:*
   $\|\delta \ell(h(x), y)\|_* \leq G, \ \forall x, y$.
6. *The sampling probability $\gamma_{k,t}$ is a valid non-zero probability: $0 < \gamma_{k,t} \leq 1$.*

With these assumptions, we show the following result. Let α_{opt} be the optimal solution.

Theorem 2. *If Properties 1 hold and $\eta = \sqrt{\frac{\sigma}{TG^2 r_\alpha}}$, then α^A, the output of* FEDBOOST *satisfies,*

$$\mathbb{E}\left[\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha_{opt})\right] \leq 2\sqrt{\frac{G^2 \sigma r_\alpha}{T}} + \frac{\alpha_* M}{2T} \sum_{t=1}^{T} \sum_{k=1}^{q} \frac{\alpha_{k,t}^2}{\gamma_{k,t}}.$$

Due to space constraints, the proof is given in Appendix B. The first $\mathcal{O}(1/\sqrt{T})$ term in the the convergence bound is similar to that of the standard mirror descent guarantees. The last term is introduced due to communication bottleneck and depends on the sampling algorithm. Observe that if we choose, uniform sampling algorithm where $\gamma_{k,t} = \frac{C}{q}$ for all k, t, then the communication dependent term becomes,

$$\sum_{t=1}^{T}\sum_{k=1}^{q}\frac{\alpha_{k,t}^2}{\gamma_{k,t}} = \sum_{t=1}^{T}\sum_{k=1}^{q}\frac{q\alpha_{k,t}^2}{C},$$

and hence is similar to applying a ℓ_2 regularization. However, note that by Cauchy-Schwarz inequality,

$$\sum_{k=1}^{q}\frac{\alpha_{k,t}^2}{\gamma_{k,t}}C \geq \left(\sum_{k=1}^{q}\alpha_{k,t}\right)^2, \tag{9}$$

and the lower bound is achieved if $\gamma_{k,t} \propto \alpha_{k,t}$. Hence to obtain the best communication efficiency, one needs to use weighted random sampling. This yields,

Corollary 1. *If Assumptions 1 hold, $\eta = \sqrt{\frac{\sigma}{TG^2 r_\alpha}}$, and $\gamma_{k,t}$ is given by (8), then α^A, the output of* FEDBOOST *satisfies,*

$$\mathbb{E}\left[\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha_{opt})\right] \leq 2\sqrt{\frac{G^2\sigma r_\alpha}{T}} + \frac{\alpha_* M}{2C}.$$

In the above analysis, the model does not converge to the true minimum due to the communication bottleneck. To overcome this, note that we can simulate a communication bandwidth of $C \cdot R$, using a communication budget of C by repeatedly doing R rounds with the same set of clients. Since, the gradients w.r.t. α only depend on the output of the predictors, it is not necessary to store all the predictors at the client at the same time. This yields the following corollary.

Corollary 2. *If Assumptions 1 hold and $\gamma_{k,t}$ is given by (8), by then by using $R = \left(\frac{\alpha_*^2 M^2 T}{C^2 G^2 \sigma r_\alpha}\right)^{1/3}$ rounds of communication with each client,*

$$\mathbb{E}\left[\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha_{opt})\right] \leq 3\left(\frac{\alpha_* M G^2 \sigma r_\alpha}{CT}\right)^{1/3}.$$

The above result has several interesting properties, First, the trade-off between convergence and communication cost is independent of the overall ensemble size q. Second, the convergence bound of $\mathcal{O}(1/T^{1/3})$ instead of the standard $\mathcal{O}(1/\sqrt{T})$ convergence bound. It is an interesting open question to determine if the above convergence bound is optimal.

4.2. Improved algorithms via bias correction

Noting that the above convergence guarantee decays dependent on $1/C$, we now show that for specific loss functions such as the ℓ_2^2 loss, we can improve the convergence result. It would be interesting to see if such results can be extended to other losses.

If the function is ℓ_2^2 loss, then for any sample x, y observe that

$$\ell(h_\alpha(x), y) = \|h_\alpha(x) - y\|_2^2 = \left\|\sum_k \alpha_k h_k(x) - y\right\|_2^2.$$

Hence,

$$\nabla_{\alpha_\ell}\ell(h_\alpha(x), y) = 2\left(\sum_k \alpha_k h_k(x) - y\right)\cdot\alpha_\ell.$$

Instead if we sample h_k with probability γ_k and use weighted random sampling as in the previous section, then the loss is

$$\ell(h_{\tilde{\alpha}}(x), y) = \left\|\sum_k \frac{\alpha_k \mathbf{1}_k}{\gamma_k} h_k(x) - y\right\|_2^2.$$

Hence,

$$\nabla\ell(h_{\tilde{\alpha}}(x), y) = 2\left(\sum_k \frac{\mathbf{1}_k\alpha_k}{\gamma_k} h_k(x) - y\right)\cdot\mathbf{1}_\ell\frac{\alpha_\ell}{\gamma_\ell}.$$

In expectation,

$$\begin{aligned}
\mathbb{E}[\nabla\ell(h_{\tilde{\alpha}}(x), y)] &= 2\,\mathbb{E}\left[\left(\sum_k \frac{\mathbf{1}_k\alpha_k}{\gamma_k} h_k(x) - y\right)\cdot\mathbf{1}_\ell\frac{\alpha_\ell}{\gamma_\ell}\right] \\
&= 2\left(\sum_{k\neq\ell}\alpha_k h_k(x) - y\right)\cdot\alpha_\ell + \left(\frac{\alpha_\ell}{\gamma_\ell}h_\ell(x) - y\right)\alpha_\ell \\
&= \nabla\ell(h_\alpha(x), y) + \alpha_\ell^2 h_\ell(x)\left(\frac{1}{\gamma_\ell} - 1\right).
\end{aligned}$$

Thus, the sampled gradients are biased. To overcome this, we propose using the following gradient estimate,

$$\nabla\ell(h_\alpha(x), y) - b, \tag{10}$$

where b is the bias correction term given by

$$b_\ell = \alpha_\ell^2 h_\ell(x)\left(\frac{1}{\gamma_\ell} - 1\right)\frac{\mathbf{1}_\ell}{\gamma_\ell}.$$

Hence in expectation,

$$\mathbb{E}[\nabla_{\alpha_\ell}\ell(h_{\tilde{\alpha}}(x), y) - b] = \nabla\ell(h_\alpha(x), y).$$

Thus the bias corrected stochastic gradient is unbiased. This gives the following corollary.

Corollary 3. *If Properties 1 holds and the loss is ℓ_2^2, then* FEDBOOST *with the bias corrected gradient (10) yields*

$$\mathbb{E}\left[\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha_{opt})\right] \leq c\sqrt{\frac{G^2\sigma r_\alpha}{T}},$$

for some constant c.

4.3. Agnostic federated ensembles

Algorithm AFLBOOST

Initialization: pre-trained $\mathbb{H} = \{h_1, ..., h_q\}$, $\lambda_1 \in \Lambda$, and $\alpha_1 = \text{argmin}_{x\in\Delta_q} F(x)$
Parameters: rounds $T \in \mathbb{Z}^+$, step size $\eta_\lambda, \eta_\alpha > 0$.
For $t = 1$ to T:

1. Uniformly sample N clients: S_t.
2. Obtain $\mathbb{H}_t$ via uniform sampling or (8).
3. For each client j:
   i. Send current ensemble model $\sum_{k\in\mathbb{H}_t} \tilde{\alpha}_{k,t} h_k$ to client j.
   ii. Obtain stochastic gradients $\nabla_\alpha \mathsf{L}_j(\tilde{\alpha}_t, \lambda_t)$, $\nabla_\lambda \mathsf{L}_j(\tilde{\alpha}_t, \lambda_t)$ and send to server.
4. $\delta_{\alpha,t}\mathsf{L} = \sum_{j\in S_t} \frac{m_j}{m} \nabla_\alpha \mathsf{L}_j(\tilde{\alpha}_t, \lambda_t)$, where $m = \sum_{j\in S_t} m_j$
5. $\delta_{\lambda,t}\mathsf{L} = \sum_{j\in S_t} \frac{m_j}{m} \nabla_\lambda \mathsf{L}_j(\tilde{\alpha}_t, \lambda_t)$
6. $v_{t+1} = [\nabla_\alpha F]^{-1}(\nabla_\alpha F(\alpha_t, \lambda_t) - \eta_\alpha \delta_{\alpha,t}\mathsf{L})$, $\alpha_{t+1} = \text{BP}(v_{t+1})$
7. $w_{t+1} = [\nabla_\lambda F]^{-1}(\nabla_\lambda F(\alpha_t, \lambda_t) + \eta_\lambda \delta_{\lambda,t}\mathsf{L})$, $\lambda_{t+1} = \text{BP}(w_{t+1})$

Output: $\alpha^A = \frac{1}{T}\sum_{t=1}^T \alpha_t$, $\lambda^A = \frac{1}{T}\sum_{t=1}^T \lambda_t$

Figure 2. Pseudocode of the AFLBOOST algorithm.

We now extend the above communication-efficient algorithm to the agnostic loss. Recall that in the agnostic loss, the optimization problem is over two sets of parameters: the ensemble weights $\alpha \in \Delta_q$ as before, and additionally the mixture weight $\lambda \in \Lambda$. We rewrite the agnostic federated losses w.r.t. these parameters using the new notation:

$$\mathsf{L}(\alpha, \lambda) = \sum_{k=1}^p \lambda_k \mathsf{L}_k(\alpha) \tag{11}$$

where $\mathsf{L}_k(\alpha)$ denotes the empirical loss of domain k as in (7). Thus, we study the following minimax optimization problem over parameters α, λ:

$$\min_{\alpha\in\Delta_q} \max_{\lambda\in\Lambda} \mathsf{L}(\alpha, \lambda).$$

The above problem can be viewed as a two player game between the server, which tries to find the best α to minimize the objective and the adversary, which maximize the objective using λ. The goal is to find the equilibrium of this minimax game, given by α_{opt} which minimizes the loss over the hardest mixture weight $\lambda_{\text{opt}} \in \Lambda$. Since ℓ is a convex function, specifically a Bregman divergence, we can approach this problem using generic mirror descent or other gradient-based instances of this algorithm.

We propose AFLBOOST, a communication-efficient stochastic ensemble algorithm that minimizes the above objective. The algorithm can be viewed as a combination of the communication-efficient approach of FEDBOOST and the stochastic mirror descent algorithm for agnostic loss (Mohri et al., 2019). To prove convergence guarantees for AFLBOOST, we need few more assumptions.

Properties 2. *Assume the following properties for the Bregman divergence defined over a function F, the loss function L, and the sets Δ_q and $\Lambda \subseteq \Delta_p$:*

1. *Let $\Lambda \subseteq \Delta_p$ is a convex, compact set.*
2. *$\lambda_* \triangleq \max_{\lambda\in\Lambda} \|\lambda\|$.*
3. *For all λ, λ', $\mathsf{B}_F(\lambda \parallel \lambda') \leq r_\lambda$.*
4. *Let $G_\alpha = \max_{\lambda,\alpha} \|\delta_\alpha \mathsf{L}\|_*$, $G_\lambda = \max_{\lambda,\alpha} \|\delta_\lambda \mathsf{L}\|_*$.*

With these assumptions, we show the following convergence guarantees for AFLBOOST with weighted random sampling. The proof is in Appendix C.

Theorem 3. *Let Properties 1 and 2 hold. Let $\eta_\lambda = \sqrt{\frac{\sigma}{TG_\lambda^2 r_\lambda}}$ and $\eta_\alpha = \sqrt{\frac{\sigma}{TG_\alpha^2 r_\alpha}}$. Let α^A be the output of* AFLBOOST. *If $\gamma_{k,t}$ is given by 8, then $\mathbb{E}[\max_{\lambda\in\Lambda} \mathsf{L}(\alpha^A, \lambda) - \min_{\alpha\in\Delta_q} \max_{\lambda\in\Lambda} \mathsf{L}(\alpha, \lambda)]$ is at most*

$$4\sqrt{\frac{G_\alpha^2(\sigma r_\alpha + \alpha_*)}{T}} + 4\sqrt{\frac{G_\lambda^2(\sigma r_\lambda + \lambda_*)}{T}} + \frac{M(\lambda_* + \alpha_*)}{C}.$$

5. Experimental validation

We demonstrate the efficacy of FEDBOOST for density estimation under various communication budgets. We compare three methods: no communication-efficiency (no sampling): $\gamma_{k,t} = 1\ \forall k, t$, uniform sampling: $\gamma = \frac{C}{q}$, and weighted random sampling: $\gamma_{k,t} \propto \alpha_{k,t} C$. For simplicity, we assume all clients participate during each round of federated training.

5.1. Synthetic dataset

We first create a synthetic dataset with $p = 100$, where each h_k is a point-mass distribution over a single element,

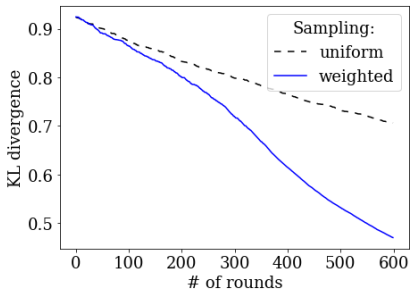

Figure 3. Comparison of loss curves for the synthetic dataset.

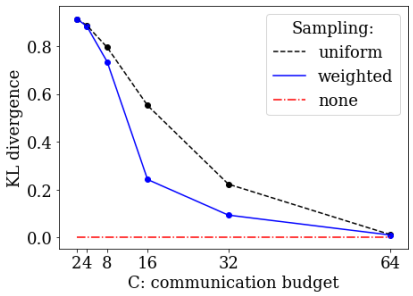

Figure 4. Comparison of sampling methods as a function of C for the synthetic dataset.

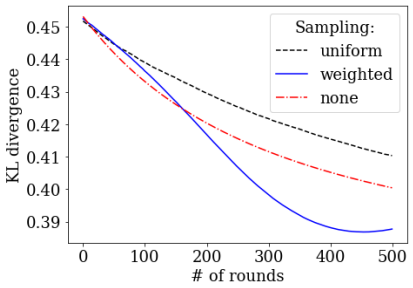

Figure 5. Comparison of loss curves for the *Shakespeare* federated learning dataset.

each α_k is initialized to $1/p$, and the true mixture weights λ follow a power law distribution. For fairness of evaluation, we fix the step size η to be 0.001 and number of rounds for both sampling methods and communication constraints, though note that this is not the ideal step size across all values of C and more optimal losses may be achieved with more extensive hyperparameter tuning. We first evaluated the results for a communication budget $C = 32$. The results are in Figure 3. As expected, the weighted sampling method performs better compared to the uniform sampling method and the loss for both methods decrease steadily.

We then compared the final loss for both uniform sampling and weighted random sampling as a function of communication budget C. The results are in Figure 4. As before, weighted random sampling performs better than uniform sampling. Furthermore, with communication budget of 64, the performance of both of them is the same as that FEDBOOST without using communication efficiency (i.e. $C = q$).

Additionally, we examine the effect of modulating the communication budget C on the rate of convergence with a larger synthetic dataset initialized in the same manner but with $p = 1000$. These results and discussion are included in the Appendix D due to space limitations.

5.2. *Shakespeare* corpus

Motivated by language modeling, we consider estimating unigram distributions for the Shakespeare TensorFlow Federated *Shakespeare* dataset, which contains dialogues in Shakespeare plays of $p = 715$ characters. We preprocessed the data by removing punctuation and converting words to lowercase. We then trained a unigram language model for each client and tried to find the best ensemble for the entire corpus using proposed algorithms (setting where $q = p$). We set $C = p/2$ and use $\eta = 0.01$. The results are in Figure 5. As before weighted random sampling performs better than uniform sampling, however somewhat surprisingly, the weighted sampling also converges better than the communication-inefficient version of FEDBOOST, which uses all base predictors at each round (i.e. $C = q$).

6. Conclusion

We proposed to learn an ensemble of pre-trained base predictors via federated learning and showed that such an ensemble based method is optimal for density estimation for both standard empirical risk minimization and agnostic risk minimization. We provided FEDBOOST and AFLBOOST, communication-efficient and theoretically-motivated ensemble algorithms for federated learning, where per-round communication cost is independent of the size of the ensemble. Finally, we empirically evaluated the proposed methods.

Acknowledgements

We warmly thank our colleagues Mingqing Chen, Rajiv Mathews, and Jae Ro for helpful discussions and comments. The work of MM was partly supported by NSF CCF-1535987, NSF IIS-1618662, and a Google Research Award.

References

Banerjee, A., Merugu, S., Dhillon, I. S., and Ghosh, J. Clustering with Bregman divergences. *Journal of machine learning research*, 6(Oct):1705–1749, 2005.

Bassily, R., Smith, A., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pp. 464–473. IEEE, 2014.

Basu, D., Data, D., Karakus, C., and Diggavi, S. Qsparse-local-sgd: Distributed sgd with quantization, sparsification and local computations. In *Advances in Neural Information Processing Systems*, pp. 14668–14679, 2019.

Bregman, L. The relaxation method of finding the com-

mon points of convex sets and its application to the solution of problems in convex programming. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967. URL https://doi.org/10.1016/0041-5553(67)90040-7.

Breiman, L. Bagging predictors. *Machine learning*, 24(2): 123–140, 1996.

Brisimi, T. S., Chen, R., Mela, T., Olshevsky, A., Paschalidis, I. C., and Shi, W. Federated learning of predictive models from federated electronic health records. *International journal of medical informatics*, 112:59–67, 2018.

Caldas, S., Konečny, J., McMahan, H. B., and Talwalkar, A. Expanding the reach of federated learning by reducing client resource requirements. *arXiv preprint arXiv:1812.07210*, 2018.

Chen, M., Mathews, R., Ouyang, T., and Beaufays, F. Federated learning of out-of-vocabulary words. *arXiv preprint arXiv:1903.10635*, 2019a.

Chen, M., Suresh, A. T., Mathews, R., Wong, A., Beaufays, F., Allauzen, C., and Riley, M. Federated learning of N-gram language models. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, 2019b.

Dietterich, T. G. An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization, Aug 2000. URL https://doi.org/10.1023/A:1007607513941.

Folland, G. B. Higher-order derivatives and taylor's formula in several variables. *Lecture notes*, 2010. URL sites.math.washington.edu/~folland/Math425/taylor2.pdf.

Freund, Y., Schapire, R., and Abe, N. A short introduction to boosting. *Journal-Japanese Society For Artificial Intelligence*, 14(771-780):1612, 1999.

Freund, Y., Mansour, Y., and Schapire, R. E. Generalization bounds for averaged classifiers. *The Annals of Statistics*, 32:1698–1722, 2004.

Gong, B., Grauman, K., and Sha, F. Connecting the dots with landmarks: Discriminatively learning domain-invariant features for unsupervised domain adaptation. In *ICML*, volume 28, pp. 222–230, 2013a.

Gong, B., Grauman, K., and Sha, F. Reshaping visual datasets for domain adaptation. In *NIPS*, pp. 1286–1294, 2013b.

Han, S., Mao, H., and Dally, W. J. Deep compression: Compressing deep neural networks with pruning, trained quantization and huffman coding. *arXiv preprint arXiv:1510.00149*, 2015.

Hard, A., Rao, K., Mathews, R., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

Hinton, G., Vinyals, O., and Dean, J. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Hoffman, J., Kulis, B., Darrell, T., and Saenko, K. Discovering latent domains for multisource domain adaptation. In *ECCV*, volume 7573, pp. 702–715, 2012.

Hoffman, J., Mohri, M., and Zhang, N. Algorithms and theory for multiple-source adaptation. In *Proceedings of NeurIPS*, pp. 8256–8266, 2018.

Jozefowicz, R., Vinyals, O., Schuster, M., Shazeer, N., and Wu, Y. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.

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

Karimireddy, S. P., Rebjock, Q., Stich, S., and Jaggi, M. Error feedback fixes signsgd and other gradient compression schemes. In *International Conference on Machine Learning*, pp. 3252–3261, 2019.

Konečný, J., McMahan, H. B., Ramage, D., and Richtárik, P. Federated optimization: Distributed machine learning for on-device intelligence. *arXiv preprint arXiv:1610.02527*, 2016a.

Konečný, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016b.

Kumar, A., Kim, J., Lyndon, D., Fulham, M., and Feng, D. An ensemble of fine-tuned convolutional neural networks for medical image classification. *IEEE journal of biomedical and health informatics*, 21(1):31–40, 2016.

Kumar, S., Nirschl, M., Holtmann-Rice, D., Liao, H., Suresh, A. T., and Yu, F. Lattice rescoring strategies for long short term memory language models in speech recognition. In *2017 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 165–172. IEEE, 2017.

MacKay, D. J. C. *Bayesian methods for adaptive models*. PhD thesis, California Institute of Technology, 1991.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Multiple source adaptation and the Rényi divergence. In *UAI*, pp. 367–374, 2009a.

Mansour, Y., Mohri, M., and Rostamizadeh, A. Domain adaptation with multiple sources. In *NIPS*, pp. 1041–1048, 2009b.

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

Mohri, M. and Medina, A. M. New analysis and algorithm for learning with drifting distributions. In *Proceedings of ALT*, pp. 124–138, 2012.

Mohri, M., Sivek, G., and Suresh, A. T. Agnostic federated learning. In *International Conference on Machine Learning*, pp. 4615–4625, 2019.

Nemirovski, A. S. and Yudin, D. B. *Problem complexity and Method Efficiency in Optimization*. Wiley, 1983.

Ramaswamy, S., Mathews, R., Rao, K., and Beaufays, F. Federated learning for emoji prediction in a mobile keyboard. *arXiv preprint arXiv:1906.04329*, 2019.

Sak, H., Senior, A., Rao, K., and Beaufays, F. Fast and accurate recurrent neural network acoustic models for speech recognition. In *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

Samarakoon, S., Bennis, M., Saad, W., and Debbah, M. Federated learning for ultra-reliable low-latency v2v communications. In *2018 IEEE Global Communications Conference (GLOBECOM)*, pp. 1–7. IEEE, 2018.

Smyth, P. and Wolpert, D. Linearly combining density estimators via stacking. *Machine Learning*, 36:59–83, July 1999.

Stich, S. U., Cordonnier, J.-B., and Jaggi, M. Sparsified SGD with memory. In *Advances in Neural Information Processing Systems*, pp. 4447–4458, 2018.

Suresh, A. T., Yu, F. X., Kumar, S., and McMahan, H. B. Distributed mean estimation with limited communication. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3329–3337. JMLR. org, 2017.

Wu, J., Leng, C., Wang, Y., Hu, Q., and Cheng, J. Quantized convolutional neural networks for mobile devices. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4820–4828, 2016.

Yang, T., Andrew, G., Eichner, H., Sun, H., Li, W., Kong, N., Ramage, D., and Beaufays, F. Applied federated learning: Improving google keyboard query suggestions. *arXiv preprint arXiv:1812.02903*, 2018.

Zhang, K., Gong, M., and Schölkopf, B. Multi-source domain adaptation: A causal view. In *AAAI*, pp. 3150–3157, 2015.

A. Proofs for density estimation

A.1. Proof of Lemma 1

Lemma 1. *Let the loss be a Bregman divergence B_F. Then, for any $\lambda \in \Lambda \subseteq \Delta_p$, if $h^* = \sum_{k=1}^p \lambda_k \mathcal{D}_k$ is in $\mathcal{H}$, then it is a minimizer of $h \mapsto \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$. If F is further strictly convex, then it is the unique minimizer.*

Proof. Fix $\lambda \in \Lambda$ such that $\sum_{k=1}^p \lambda_k \mathcal{D}_k$ is in $\mathcal{H}$. By the non-negativity of the Bregman divergence, for all h, $\mathsf{B}_F(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel h) \geq 0$ and equality is achieved for $h = \sum_{k=1}^p \lambda_k \mathcal{D}_k$. Thus, h^* is a minimizer of $h \mapsto \mathsf{B}_F(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel h)$. Since F is strictly convex, $h \mapsto \mathsf{B}_F(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel h)$ is strictly convex and h^* is therefore the unique minimizer.

Now, for any hypothesis h, observe that the following difference is a constant independent of h:

$$\begin{aligned}
&\sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) - \mathsf{B}_F\left(\sum_{k=1}^p \lambda_k \mathcal{D}_k \parallel h\right) \\
&= \sum_{k=1}^p \lambda_k \left[F(\mathcal{D}_k) - F(h) - \langle \nabla F(h), \mathcal{D}_k - h\rangle\right] - \left[F\left(\sum_{k=1}^p \lambda_k \mathcal{D}_k\right) - F(h) - \left\langle \nabla F(h), \sum_{k=1}^p \lambda_k \mathcal{D}_k - h\right\rangle\right] \\
&= \sum_{k=1}^p \lambda_k F(\mathcal{D}_k) - F\left(\sum_{k=1}^p \lambda_k \mathcal{D}_k\right).
\end{aligned} \tag{12}$$

Thus, h^* is also the unique minimizer of $h \mapsto \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$. □

A.2. Proof of Lemma 2

Lemma 2. *Let the loss be a Bregman divergence B_F with F strictly convex and assume that $\operatorname{conv}(\{\mathcal{D}_1, ..., \mathcal{D}_p\}) \subseteq \mathcal{H}$. Observe that B_F is jointly convex in both arguments. Then, for any convex set $\Lambda \subseteq \Delta_p$, the solution of the optimization problem $\min_{h\in\mathcal{H}} \max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ exists and is in $\operatorname{conv}(\{\mathcal{D}_1, ..., \mathcal{D}_p\})$.*

Proof. Let $\mathcal{H}'$ is the closure of convex hull of $\mathcal{H}$. Observe that $\mathcal{H}'$ is a convex and compact set.

$$\min_{h\in\mathcal{H}'} \max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) \leq \min_{h\in\mathcal{H}} \max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big).$$

We show that minimizer over $\mathcal{H}'$ exists and is in the $\operatorname{conv}(\{\mathcal{D}_1, \ldots, \mathcal{D}_p\})$. Since $\operatorname{conv}(\{\mathcal{D}_1, \ldots, \mathcal{D}_p\}) \subseteq \mathcal{H} \subseteq \mathcal{H}'$, the minimizer over $\mathcal{H}$ also exists and is in the $\operatorname{conv}(\{\mathcal{D}_1, \ldots, \mathcal{D}_p\})$.

Since B_F is convex with respect to its second argument, $h \mapsto \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ is a convex function of h defined over the convex set $\mathcal{H}'$. Since any maximum of a convex function is also convex, $h \mapsto \max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ is a convex function and its minimum over the compact set $\mathcal{H}'$ exists.

We now show that the minimizer is in $\operatorname{conv}(\{\mathcal{D}_1, \ldots, \mathcal{D}_p\})$. Notice that, since $\sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ is linear in λ, we have

$$\max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big) = \max_{\lambda\in\operatorname{conv}(\Lambda)} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big).$$

Thus, it suffices to consider the case $\Lambda \subseteq \Delta_p$. Then, since $\mathcal{H}'$ is a compact and convex set and since B_F is convex with respect to its second argument, by Sion's minimax theorem, we can write:

$$\min_{h\in\mathcal{H}'} \max_{\lambda\in\Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right) = \max_{\lambda\in\Lambda} \min_{h\in\mathcal{H}'} \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right).$$

Let $\lambda^{\text{opt}} = \operatorname{argmax}_{\lambda\in\Lambda} \min_{h\in\mathcal{H}'} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ and $h^* = \sum_k \lambda_k^{\text{opt}} \mathcal{D}_k$. By assumption, $\operatorname{conv}(\{\mathcal{D}_1, \ldots, \mathcal{D}_p\})$ is included in $\mathcal{H}'$, thus h^* is in $\mathcal{H}'$ and, by Lemma 1, h^* is a minimizer of $h \mapsto \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F(\mathcal{D}_k \parallel h)$. In view of that, if h'

is a minimizer of $h \mapsto \max_{\lambda \in \Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ over $\mathcal{H}'$, then the following holds:

$$\begin{aligned}
\max_{\lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h'\right) &\geq \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F\left(\mathcal{D}_k \parallel h'\right) && \text{(def. of max)} \\
&\geq \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F\left(\mathcal{D}_k \parallel h^*\right) && \text{(Lemma 1)} \\
&= \min_{h \in \mathcal{H}'} \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right) && (h^* \text{ minimizer}) \\
&= \max_{\lambda \in \Lambda} \min_{h \in \mathcal{H}'} \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right) && (\text{def. of } \lambda_k^{\text{opt}}) \\
&= \min_{h \in \mathcal{H}'} \max_{\lambda \in \Lambda} \sum_{k=1}^p \lambda_k \mathsf{B}_F\left(\mathcal{D}_k \parallel h\right). && \text{(Sion's minimax theorem)}
\end{aligned}$$

By the optimality of h', the first and last expressions in this chain of inequalities are equal, which implies the equality of all intermediate terms. In particular, this implies $\sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F(\mathcal{D}_k \parallel h') = \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F(\mathcal{D}_k \parallel h^*)$. Since F is strictly convex, by Lemma 1, the minimizer of $h \mapsto \sum_{k=1}^p \lambda_k^{\text{opt}} \mathsf{B}_F\big(\mathcal{D}_k \parallel h\big)$ is unique and $h' = h^*$. This completes the proof. □

B. Convergence guarantee of FEDBOOST (Theorem 2)

Theorem 2. *If Properties 1 hold and* $\eta = \sqrt{\frac{\sigma}{TG^2 r_\alpha}}$*, then* α^A*, the output of* FEDBOOST *satisfies,*

$$\mathbb{E}\left[\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha_{opt})\right] \leq 2\sqrt{\frac{G^2 \sigma r_\alpha}{T}} + \frac{\alpha_* M}{2T} \sum_{t=1}^T \sum_{k=1}^q \frac{\alpha_{k,t}^2}{\gamma_{k,t}}.$$

Proof. By Jensen's inequality,

$$\mathsf{L}(\alpha^A) \leq \frac{1}{T} \sum_{t=1}^T \mathsf{L}(\alpha_t).$$

Hence, it suffices to bound

$$\frac{1}{T} \sum_{t=1}^T \left(\mathsf{L}(\alpha_t) - \mathsf{L}(\alpha)\right).$$

For any t,

$$\begin{aligned}
\mathsf{L}(\alpha_t) - \mathsf{L}(\alpha) &\leq \langle \nabla \mathsf{L}(\alpha_t), \alpha_t - \alpha \rangle && \text{(convexity of L)} \\
&= \langle \delta_t \mathsf{L}, \alpha_t - \alpha \rangle + \langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha \rangle \\
&= \frac{1}{\eta} \langle \nabla F(\alpha_t) - \nabla F(v_{t+1}), \alpha_t - \alpha \rangle + \langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha \rangle && \text{(def. of } v_{t+1}) \\
&= \frac{1}{\eta} \left(\mathsf{B}_F(\alpha \parallel \alpha_t) + \mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha \parallel v_{t+1})\right) + \langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha \rangle && \text{(Bregman div. def.)} \\
&\leq \frac{1}{\eta} \left(\mathsf{B}_F(\alpha \parallel \alpha_t) + \mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha \parallel \alpha_{t+1}) - \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1})\right) && \text{(13a)} \\
&+ \langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha \rangle, && \text{(13b)}
\end{aligned}$$

where the last inequality follows because $\mathsf{B}_F(\alpha \parallel v_{t+1}) \geq \mathsf{B}_F(\alpha \parallel \alpha_{t+1}) + \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1})$ by the generalized

Pythagorean inequality. For the first term (13a), summing over t gives the following telescoping sum,

$$\begin{aligned}\sum_{t=1}^{T}(\mathsf{B}_F(\alpha \parallel \alpha_t) + \mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha \parallel \alpha_{t+1}) - \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1})) \qquad (14)\\ = \mathsf{B}_F(\alpha \parallel \alpha_1) - \mathsf{B}_F(\alpha \parallel \alpha_{T+1}) + \sum_{t=1}^{T} \mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1})\\ \leq \mathsf{B}_F(\alpha \parallel \alpha_1) + \sum_{t=1}^{T} (\mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1}))\,.\end{aligned}$$

Now consider the summation term:

$$\begin{aligned}\mathsf{B}_F(\alpha_t \parallel v_{t+1}) - \mathsf{B}_F(\alpha_{t+1} \parallel v_{t+1}) &= F(\alpha_t) - F(\alpha_{t+1}) - \langle \nabla F(v_{t+1}), \alpha_t - \alpha_{t+1}\rangle\\ &\leq \langle \nabla F(\alpha_t), \alpha_t - \alpha_{t+1}\rangle - \frac{\sigma}{2}\|\alpha_t - \alpha_{t+1}\|^2 - \langle \nabla F(v_{t+1}), \alpha_t - \alpha_{t+1}\rangle \qquad \text{(strong convexity of } F)\\ &= \langle \nabla F(\alpha_t) - \nabla F(v_{t+1}), \alpha_t - \alpha_{t+1}\rangle - \frac{\sigma}{2}\|\alpha_t - \alpha_{t+1}\|^2\\ &= \eta\langle \delta_t \mathsf{L}, \alpha_t - \alpha_{t+1}\rangle - \frac{\sigma}{2}\|\alpha_t - \alpha_{t+1}\|^2 \qquad \text{(def. of } v_{t+1})\\ &\leq \eta\|\delta_t \mathsf{L}\|_*\|\alpha_t - \alpha_{t+1}\| - \frac{\sigma}{2}\|\alpha_t - \alpha_{t+1}\|^2 \qquad \text{(Cauchy-Schwarz ineq.)}\\ &\leq \frac{\eta^2\|\delta_t \mathsf{L}\|_*^2}{2\sigma}. \qquad (15)\end{aligned}$$

Combining the above inequalities,

$$\begin{aligned}\sum_{t=1}^{T}(\mathsf{L}(\alpha_t) - \mathsf{L}(\alpha)) &\leq \frac{1}{\eta}\mathsf{B}_F(\alpha \parallel \alpha_1) + \sum_{t=1}^{T}\left(\frac{\eta\|\delta_t \mathsf{L}\|_*^2}{2\sigma} + \langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha\rangle\right)\\ &\leq \frac{1}{\eta}\mathsf{B}_F(\alpha \parallel \alpha_1) + \frac{\eta G^2 T}{2\sigma} + \sum_{t=1}^{T}(\langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha\rangle)\,.\end{aligned}$$

We now bound (13b) in expectation, the inner product term in the above equation. Denote by $\nabla_t \mathsf{L}(\cdot) := \sum_{j\in S_t}\frac{m_j}{m}\nabla \mathsf{L}_j(\cdot)$, where $m = \sum_{j\in S_t} m_j$. Taking the expectation over $j \in S_t$,

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T}\langle \nabla \mathsf{L}(\alpha_t) - \delta_t \mathsf{L}, \alpha_t - \alpha\rangle\right] &= \sum_{t=1}^{T}\langle \nabla \mathsf{L}(\alpha_t) - \mathbb{E}\left[\delta_t \mathsf{L}\right], \alpha_t - \alpha\rangle \qquad (16)\\ &= \sum_{t=1}^{T}\langle \nabla \mathsf{L}(\alpha_t) - \mathbb{E}\left[\nabla_t \mathsf{L}(\tilde{\alpha}_t)\right], \alpha_t - \alpha\rangle\\ &\leq \sum_{t=1}^{T}\|\nabla \mathsf{L}(\alpha_t) - \mathbb{E}\left[\nabla_t \mathsf{L}(\tilde{\alpha}_t)\right]\|_*\|\alpha_t - \alpha\| \qquad \text{(Cauchy-Schwarz ineq.)}\\ &\leq \sum_{t=1}^{T}\|\nabla \mathsf{L}(\alpha_t) - \mathbb{E}\left[\nabla_t \mathsf{L}(\tilde{\alpha}_t)\right]\|_*\alpha_*. \qquad \text{(by Prop. 1.2.)}\\ & \qquad (17)\end{aligned}$$

To understand $\mathbb{E}[\nabla_t \mathsf{L}(\tilde{\alpha}_t)]$, we use Taylor's Theorem in several variables (Folland, 2010). Let $f = \nabla_t \mathsf{L}$. Expanding $f(\tilde{\alpha}_t)$ about α_t,

$$f(\tilde{\alpha}_t) - f(\alpha_t) = \nabla f(\alpha_t)(\tilde{\alpha}_t - \alpha_t) + R_1(\tilde{\alpha}_t - \alpha_t),$$

where $R_1(\cdot)$ is the reminder term that can be bounded as

$$\|R_1(\tilde{\alpha}_t - \alpha_t)\|_* \leq \frac{M}{2!}\|\tilde{\alpha}_t - \alpha_t\|_2^2,$$

with $\|\nabla^2 f(\cdot)\| \leq M$. Taking expectation over $\tilde{\alpha}_t$ and using the fact that $\mathbb{E}[\tilde{\alpha}_t] = \alpha_t$, we get

$$\begin{aligned}
\|\,\mathbb{E}[f(\tilde{\alpha}_t)] - f(\alpha_t)\| &\leq \frac{M}{2}\,\mathbb{E}\,\|\tilde{\alpha}_t - \alpha_t\|_2^2 \\
&= \frac{M}{2}\sum_{k=1}^{q}\mathbb{E}\left\|\frac{\alpha_{k,t}\mathbf{1}_{k,t}}{\gamma_{k,t}} - \alpha_{k,t}\right\|_2^2 \\
&= \frac{M}{2}\sum_{k=1}^{q}\alpha_{k,t}^2\,\mathbb{E}\left\|\frac{\mathbf{1}_{k,t}}{\gamma_{k,t}} - 1\right\|_2^2 \\
&= \frac{M}{2}\sum_{k=1}^{q}\alpha_{k,t}^2\left(\frac{1-\gamma_{k,t}}{\gamma_{k,t}}\right) \\
&\leq \frac{M}{2}\sum_{k=1}^{q}\frac{\alpha_{k,t}^2}{\gamma_{k,t}}.
\end{aligned}$$

Combining the resulting inequalities gives

$$\mathsf{L}(\alpha^A) - \mathsf{L}(\alpha) \leq \frac{1}{\eta T}\mathsf{B}_F(\alpha \parallel \alpha_1) + \frac{\eta G^2}{2\sigma} + \frac{\alpha_* M}{2T}\sum_{t=1}^{T}\sum_{k=1}^{q}\frac{\alpha_{k,t}^2}{\gamma_{k,t}}.$$

Choosing the learning rate yields the theorem. □

C. Convergence guarantee for AFLBOOST (Theorem 3)

Theorem 3. *Let Properties 1 and 2 hold. Let* $\eta_\lambda = \sqrt{\frac{\sigma}{TG_\lambda^2 r_\lambda}}$ *and* $\eta_\alpha = \sqrt{\frac{\sigma}{TG_\alpha^2 r_\alpha}}$. *Let* α^A *be the output of* AFLBOOST. *If* $\gamma_{k,t}$ *is given by 8, then* $\mathbb{E}[\max_{\lambda\in\Lambda}\mathsf{L}(\alpha^A,\lambda) - \min_{\alpha\in\Delta_q}\max_{\lambda\in\Lambda}\mathsf{L}(\alpha,\lambda)]$ *is at most*

$$4\sqrt{\frac{G_\alpha^2(\sigma r_\alpha + \alpha_*)}{T}} + 4\sqrt{\frac{G_\lambda^2(\sigma r_\lambda + \lambda_*)}{T}} + \frac{M(\lambda_* + \alpha_*)}{C}.$$

Proof. By Mohri et al. (2019)[Lemma 5], it suffices to bound

$$\frac{1}{T}\max_{\substack{\lambda\in\Lambda;\\ \alpha\in\Delta_q}}\{\sum_{t=1}^{T}\mathsf{L}(\alpha_t,\lambda) - \mathsf{L}(\alpha,\lambda_t)\}. \tag{18}$$

Consider the following inequalities:

$$\begin{aligned}
\mathsf{L}(\alpha_t,\lambda) - \mathsf{L}(\alpha,\lambda_t) &= \mathsf{L}(\alpha_t,\lambda) - \mathsf{L}(\alpha_t,\lambda_t) + \mathsf{L}(\alpha_t,\lambda_t) - \mathsf{L}(\alpha,\lambda_t) \\
&\leq \langle\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t), \lambda - \lambda_t\rangle + \langle\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t), \alpha_t - \alpha\rangle \qquad \text{(convexity of } \mathsf{L}\text{)} \\
&= \langle\delta_{\lambda,t}\mathsf{L}, \lambda - \lambda_t\rangle + \langle\delta_{\alpha,t}\mathsf{L}, \alpha_t - \alpha\rangle \\
&\quad + \langle\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t) - \delta_{\lambda,t}\mathsf{L}, \lambda - \lambda_t\rangle + \langle\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t) - \delta_{\alpha,t}\mathsf{L}, \alpha_t - \alpha\rangle
\end{aligned}$$

Given these inequalities, we can bound (18) using the sub-additive property of $\max$ on the previous inequality as follows:

$$\max_{\substack{\lambda\in\Lambda;\\ \alpha\in\Delta_q}}\{\sum_{t=1}^{T}\mathsf{L}(\alpha_t,\lambda)-\mathsf{L}(\alpha,\lambda_t)\}$$

$$\leq \max_{\substack{\lambda\in\Lambda;\\ \alpha\in\Delta_q}}\sum_{t=1}^{T}\{\langle\delta_{\lambda,t}\mathsf{L},\lambda-\lambda_t\rangle+\langle\delta_{\alpha,t}\mathsf{L},\alpha_t-\alpha\rangle\} \tag{19a}$$

$$+\max_{\substack{\lambda\in\Lambda;\\ \alpha\in\Delta_q}}\sum_{t=1}^{T}\{\langle\lambda,\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\lambda,t}\mathsf{L}\rangle+\langle\alpha,\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\alpha,t}\mathsf{L}\rangle\} \tag{19b}$$

$$+\sum_{t=1}^{T}\langle\lambda_t,\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\lambda,t}\mathsf{L}\rangle+\langle\alpha_t,\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\alpha,t}\mathsf{L}\rangle, \tag{19c}$$

which we will bound in three parts. Consider the first sub-equation (19a): similarly in arriving at (13a), it follows by definition of w_{t+1}, v_{t+1} that

$$\langle\delta_{\lambda,t}\mathsf{L},\lambda-\lambda_t\rangle+\langle\delta_{\alpha,t}\mathsf{L},\alpha_t-\alpha\rangle\leq\frac{1}{\eta_\lambda}(\mathsf{B}_F(\lambda\parallel\lambda_t)+\mathsf{B}_F(\lambda_t\parallel w_{t+1})-\mathsf{B}_F(\lambda\parallel\lambda_{t+1})-\mathsf{B}_F(\lambda_{t+1}\parallel w_{t+1}))$$
$$+\frac{1}{\eta_\alpha}(\mathsf{B}_F(\alpha\parallel\alpha_t)+\mathsf{B}_F(\alpha_t\parallel v_{t+1})-\mathsf{B}_F(\alpha\parallel\alpha_{t+1})-\mathsf{B}_F(\alpha_{t+1}\parallel v_{t+1})).$$

Summing over t, this gives the following by similar argument as in (14) for all λ,α:

$$\sum_{t=1}^{T}\langle\delta_{\lambda,t}\mathsf{L},\lambda-\lambda_t\rangle+\langle\delta_{\alpha,t}\mathsf{L},\alpha_t-\alpha\rangle\leq\frac{1}{\eta_\lambda}(\mathsf{B}_F(\lambda\parallel\lambda_1)+\sum_{t=1}^{T}\mathsf{B}_F(\lambda_t\parallel w_{t+1})-\mathsf{B}_F(\lambda_{t+1}\parallel w_{t+1}))$$
$$+\frac{1}{\eta_\alpha}(\mathsf{B}_F(\alpha\parallel\alpha_1)+\sum_{t=1}^{T}\mathsf{B}_F(\alpha_t\parallel v_{t+1})-\mathsf{B}_F(\alpha_{t+1}\parallel v_{t+1}))$$

In view of the inequality resulting from (15), for all λ,α, this is bounded by

$$\sum_{t=1}^{T}\langle\delta_{\lambda,t}\mathsf{L},\lambda-\lambda_t\rangle+\langle\delta_{\alpha,t}\mathsf{L},\alpha_t-\alpha\rangle\leq\frac{1}{\eta_\lambda}\mathsf{B}_F(\lambda\parallel\lambda_1)+\frac{1}{\eta_\alpha}\mathsf{B}_F(\alpha\parallel\alpha_1)+\sum_{t=1}^{T}\frac{\eta_\lambda^2\|\delta_{\lambda,t}\mathsf{L}\|_*^2+\eta_\alpha^2\|\delta_{\alpha,t}\mathsf{L}\|_*^2}{2\sigma}$$
$$=\frac{1}{\eta_\lambda}\mathsf{B}_F(\lambda\parallel\lambda_1)+\frac{1}{\eta_\alpha}\mathsf{B}_F(\alpha\parallel\alpha_1)+\frac{T\left(\eta_\lambda G_\lambda^2+\eta_\alpha G_\alpha^2\right)}{2\sigma}.$$

Next, we proceed with the bound for third sub-equation (19c) in expectation via similar argument followed to arrive at (15):

$$\mathbb{E}[\sum_{t=1}^{T}\langle\lambda_t,\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\lambda,t}\mathsf{L}\rangle+\langle\alpha_t,\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\alpha,t}\mathsf{L}\rangle]$$
$$=\sum_{t=1}^{T}\langle\lambda_t,\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t)-\mathbb{E}[\nabla_{t,\lambda}\mathsf{L}(\tilde{\alpha}_t,\lambda_t)]\rangle+\langle\alpha_t,\nabla_\alpha\mathsf{L}(\tilde{\alpha}_t,\lambda_t)-\mathbb{E}[\nabla_{t,\alpha}\mathsf{L}(\tilde{\alpha}_t,\lambda_t)]\rangle,$$

where $\nabla_{t,\lambda}\mathsf{L}(\cdot):=\sum_{j\in S_t}\frac{m_j}{m}\nabla_\lambda\mathsf{L}_j(\cdot)$, and similarly for $\nabla_{t,\alpha}\mathsf{L}(\cdot)$. Similar to the proof of (15) and (9), it can be shown that

$$\sum_{t=1}^{T}\langle\lambda_t,\nabla_\lambda\mathsf{L}(\alpha_t,\lambda_t)-\mathbb{E}[\nabla_{t,\lambda}]\rangle\leq\frac{MT\lambda_*}{2C}.$$

Similarly,

$$\mathbb{E}[\sum_{t=1}^{T}\langle\alpha_t,\nabla_\alpha\mathsf{L}(\alpha_t,\lambda_t)-\delta_{\alpha,t}\mathsf{L}\rangle]\leq\frac{MT\alpha_*}{2C}.$$

Combining the two bounds, we have

$$\mathbb{E}[\sum_{t=1}^{T}\langle\lambda_t, \nabla_\lambda \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\lambda,t}\mathsf{L}\rangle + \langle\alpha_t, \nabla_\alpha \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\alpha,t}\mathsf{L}\rangle] \leq \frac{MT(\alpha_* + \lambda_*)}{2C}.$$

We now consider the second sub-equation term (19b), focusing on the first summand with the max over λ and bound this by the Cauchy-Schwarz inequality, then Jensen's inequality:

$$\begin{aligned}
&\mathbb{E}[\max_{\lambda\in\Lambda}\{\sum_{t=1}^{T}\langle\lambda, \nabla_\lambda \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\lambda,t}\mathsf{L}\rangle\}] \\
&\leq \mathbb{E}[\max_{\lambda\in\Lambda}\{\sum_{t=1}^{T}\langle\lambda, \nabla_\lambda \mathsf{L}(\alpha_t,\lambda_t) - \mathbb{E}[\delta_{\lambda,t}\mathsf{L}]\rangle\}] + \mathbb{E}[\max_{\lambda\in\Lambda}\{\sum_{t=1}^{T}\langle\lambda, \delta_{\lambda,t}\mathsf{L} - \mathbb{E}[\delta_{\lambda,t}\mathsf{L}]\rangle\}] \\
&\leq \frac{MT\lambda_*}{2C} + \lambda_* G_\lambda\sqrt{T},
\end{aligned}$$

where λ_* denotes the $\max$ over the compact set Λ. Similarly, we can obtain the following inequality:

$$\mathbb{E}[\max_{\alpha\in\Delta_q}\sum_{t=1}^{T}\langle\alpha, \nabla_\alpha \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\alpha,t}\mathsf{L}\rangle] \leq \frac{MT\alpha_*}{2C} + \alpha_* G_\alpha\sqrt{T}.$$

Thus, combining the inequalities gives

$$\max_{\substack{\lambda\in\Lambda;\\ \alpha\in\Delta_q}}\sum_{t=1}^{T}\{\langle\lambda, \nabla_\lambda \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\lambda,t}\mathsf{L}\rangle + \langle\alpha, \nabla_\alpha \mathsf{L}(\alpha_t,\lambda_t) - \delta_{\alpha,t}\mathsf{L}\rangle\} = \frac{MT(\lambda_* + \alpha_*)}{2C} + \alpha_* G_\alpha\sqrt{T} + \lambda_* G_\lambda\sqrt{T}.$$

Combining the bounds for (19a), (19b), and (19c), the following bound holds:

$$\begin{aligned}
&\frac{1}{T}\sum_{t=1}^{T}\eta_\alpha \mathsf{L}(\alpha_t,\lambda) - \eta_\lambda \mathsf{L}(\alpha,\lambda_t) \\
&\qquad\leq \frac{1}{T}\left(\frac{\mathsf{B}_F(\lambda \parallel \lambda_1)}{\eta_\lambda} + \frac{\mathsf{B}_F(\alpha \parallel \alpha_1)}{\eta_\alpha}\right) + \frac{T\left(\eta_\lambda G_\lambda^2 + \eta_\alpha G_\alpha^2\right)}{2\sigma} + \frac{M(\lambda_* + \alpha_*)}{C} + \frac{\alpha_* G_\alpha + \lambda_* G_\lambda}{\sqrt{T}}.
\end{aligned}$$

□

D. Additional density estimation experiments on synthetic data

Continuing the experimental validation of FEDBOOST as described in 5.1, we examine the effect of modulating the communication budget C on a density estimation task using the same setup as before, but with a power-law distributed synthetic dataset with parameter $p = 1000$.

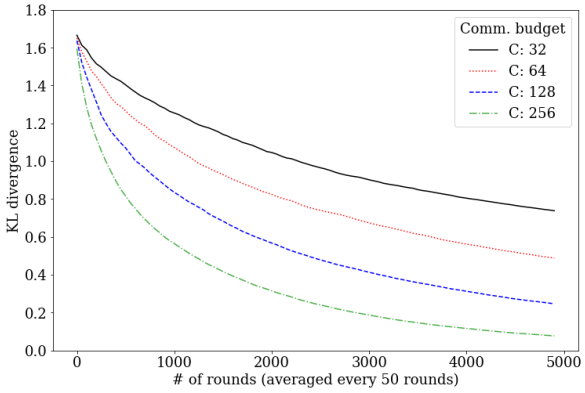

Figure 6. Comparison of loss curves as a function of C using *uniform sampling* in density estimation on synthetic data.

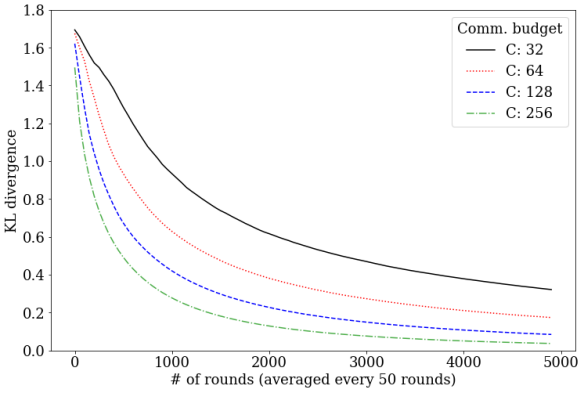

Figure 7. Comparison of convergence as C varies using *weighted random sampling* for density estimation.

We use a step size of $\eta = 0.001$ for all values of C, and include ℓ_1 regularization in the experiment using *weighted random sampling* (Fig. 7). The experimental setup is otherwise the same for both Fig. 6 and 7. Across all values of C, *weighted random sampling* of the h_k achieves lower loss than using *uniform sampling*, which validates that using weighted random sampling reduces the communication-dependent term of FEDBOOST.