FEDERATED LEARNING WITH L_0 CONSTRAINT VIA PROBABILISTIC GATES FOR SPARSITY

Anonymous authors

000

001

002003004

010 011

012

013

014

015

016

017

018

019

020

021

022

023

024

025

026

027

028

031

033

037

040

041

042

043 044

046

047

048

051 052 Paper under double-blind review

ABSTRACT

Federated Learning (FL) is a distributed machine learning setting that requires multiple clients to collaborate on training a model while maintaining data privacy. The unaddressed inherent sparsity in data often results in overly dense models and poor generalizability under data and client participation heterogeneity. We propose FL with an L_0 constraint on the density of non-zero parameters, achieved through a reparameterization using probabilistic gates and their continuous relaxation: originally proposed for sparsity in centralized machine learning. We show that the objective for L_0 constrained stochastic minimization naturally arises from an entropy maximization problem of the stochastic gates and propose an algorithm based on federated stochastic gradient descent for distributed learning. We demonstrate that the target density (ρ) of parameters can be achieved in FL, under data and client participation heterogeneity, with minimal loss in statistical performance for linear models: (i) Linear regression (LR). (ii) Logistic regression (LG). (iii) Softmax multi-class classification (MC). (iv) Multi-label classification with logistic units (MLC), and compare the results with a magnitude pruning-based algorithm for sparsity in FL. Experiments on synthetic data with target density down to $\rho = 5\%$ and publicly available e2006-tfidf, RCV1, and MNIST datasets with target density down to $\rho = 0.5\%$ demonstrate that our approach consistently works better in both sparsity recovery and statistical performance.

1 Introduction

FL training algorithms are defined by the requirements of data privacy and the distributed nature of learning algorithms (McMahan et al., 2017). The optimization in FL is challenging due to data heterogeneity and the availability of clients or devices, making the averaging of models or gradients inefficient. FL systems also eliminate the need for centralization of data, even in cases where there is no privacy concern (Kairouz et al., 2021). The resources available at devices participating in FL vary across different settings. In cross-device FL, where edge devices are often hardware-restricted, reducing computational and communication overheads, either in training or inference, is beneficial (Wang et al., 2021). A sparsity-inducing learning methodology is desirable to meet the system requirements. A sparse model, as opposed to an overly trained dense model, is favored in machine learning (ML) for its generalizability (Tibshirani, 1996). Previous works include sparse regression (Bertsimas et al., 2020), and different algorithms on Lasso sparse regression (Frandi et al., 2016; Šehić et al., 2022).

The standard approach for achieving sparsity is to utilize the L_p norm for regularization. The traditional Ridge (L_2) and Lasso (L_1) penalties depend on the magnitude of the weights, resulting in varying levels of shrinkage. In contrast, the L_0 regularizer has a constant penalty for non-zero parameters, making it a magnitude-independent penalizer. For this reason, we prefer to apply an L_0 constraint in FL to learn a global model with the desired parameter density (ρ) . Consider C clients in FL holding data $(D^{(c)})_{c=1}^C: (X^c, Y^c)$ where $X^c \in \mathbb{R}^{n_c \times p}, Y^c \in \mathbb{R}^{n_c}$ and $\sum_{c=1}^C n_c = N$. Assuming a linear model $h(x,\theta): \mathbb{R}^p \to \mathbb{R}$ and loss $\ell(h(x;\theta),y)$ where $x \in \mathbb{R}^p,y \in \mathbb{R}$, and $\theta \in \mathbb{R}^p$, the L_0 density constrained minimization problem and the associated Lagrangian can be written as

$$\min_{\theta} \quad \sum_{c=1}^{C} \frac{n_c}{N} \mathcal{L}^{(c)}(\theta) \quad \text{subject to} \quad \frac{\|\theta\|_0}{|\theta|} \le \rho, \quad \|\theta\|_0 = \sum_{j=1}^{|\theta|} \mathbb{I}[\theta_j \ne 0], \text{ and}$$
 (1)

$$\mathfrak{L}(\theta, \lambda) = \sum_{c=1}^{C} \frac{n_c}{N} \mathcal{L}^{(c)}(\theta) + \lambda \left(\frac{\|\theta\|_0}{|\theta|} - \rho \right). \tag{2}$$

Here, $\mathcal{L}^{(c)}(\theta)$ is a normalized loss at client c evaluated as

$$\mathcal{L}^{(c)}(\theta) = \frac{1}{n_c} \sum_{i=1}^{n_c} \ell(h(x_i^c; \theta), x_i^c)).$$
 (3)

A min-max problem of Lagrangian associated with a constrained optimization problem can generally be solved using gradient descent-ascent, but the presence of a non-differentiable L_0 pseudonorm poses a challenge for gradient descent-based optimization, a convenient choice for training ML models.

Louizos et al. (2017) present an L_0 regularized objective with a set of gates $z \in \mathbb{R}^p$ introduced in a reparameterization of $\theta = \tilde{\theta} \odot z^1$. Assuming a Hard concrete distribution for sampling gates z, the L_0 pseudo norm can be approximated as the sum of probabilities of gates being active $(P(z_j=1))$, enabling the application of the gradient-descent method for the minimization of L_0 regularized objective. This method allows passing information about the desired sparsity target through initialization and regularization strength; however, tuning the regularization coefficient to achieve the desired sparsity can be challenging. Gallego-Posada et al. (2022) utilize the same approach, except that they use an L_0 density constrained formulation and the min-max problem of the associated Lagrangian. We adopt this approach for our optimization problem in FL due to its flexibility in accommodating generic loss functions and learning controlled sparsity during training through simultaneous gradient descent and ascent.

In this work, we show that the same Lagrangian formulation for the L_0 constrained problem in centralized machine learning can be derived from the entropy maximization of the stochastic gates. We then propose a distributed algorithm for learning a sparse global model by aggregating reparameterized gradients across clients. Experiments on synthetic data are used to test the sparsity recovery and statistical performance, as well as the statistical performance at the desired parameter density in real-world datasets. All experiments are conducted on data distributed heterogeneously across clients by design, and with simulation of stagglers or client participation heterogeneity in the training algorithm.

The latter part of the paper is organized into sections on L_0 - constrained formulation, followed by a distributed algorithm, experiments, and a discussion.

$2 \quad L_0$ constrained formulation

2.1 CENTRALIZED ML

Assuming a linear model $h(x,\theta): \mathbb{R}^p \to \mathbb{R}$ and loss $\ell(h(x;\theta),y)$ where $x \in \mathbb{R}^p, y \in \mathbb{R}$, and $\theta \in \mathbb{R}^p$, with centralized data $D: (X,Y), X \in \mathbb{R}^{N \times p}$, and $Y \in \mathbb{R}^N$. The L_0 density-constrained minimization problem with a desired density of ρ can be described as shown below in a centralized setting.

$$\min_{\theta} \quad \frac{1}{N} \sum_{i=1}^{N} \ell(h(x_i; \theta), y_i) \quad \text{subject to} \quad \frac{\|\theta\|_0}{|\theta|} \le \rho, \quad \|\theta\|_0 = \sum_{j=1}^{|\theta|} \mathbb{I}[\theta_j \ne 0]$$
 (4)

Since the presence of a non-differentiable pseudo-norm makes the application of gradient descent-based methods infeasible on the min-max problem of the Lagrangian, Gallego-Posada et al. (2022) use an alternate formulation with reparameterization of $\theta(\tilde{\theta} \odot z)$ using gates $z \in [0,1]^p$ with hard concrete distribution (Louizos et al., 2017) for z_j with parameters ϕ_j . The minimization objective is an expectation of the loss with respect to the distribution of gates, and the sum of the probabilities of stochastic gates being non-zero is the continuous approximation of the L_0 pseudo-norm. z_j being a deterministic transformation of a parameter-free noise allows for a joint optimization of the Monte Carlo approximation of the expected loss over the noise, with respect to $\tilde{\theta}$ and ϕ using

¹⊙ stands for the Hadamard product (Horn, 1990)

reparameterized gradients (Ranganath, 2017, ch. 3.3.3). The Lagrangian and the min-max problem associated with the constrained optimization problem with this reparameterization are

$$\hat{\mathfrak{L}}(\tilde{\theta}, \phi; \{\lambda\}) = \sum_{r=1}^{R} \frac{1}{R} \left[\frac{1}{N} \sum_{i=1}^{N} \mathcal{L}\left(h(x_i; \tilde{\theta} \odot z^{(r)}), y_i\right) \right] + \lambda \left[\sum_{j=1}^{|\theta|} \frac{E_{q(z|\phi)}[z_j]}{|\theta|} - \rho \right], \text{ and} \quad (5)$$

$$\theta^*, \phi^*, \lambda^* = \arg\min_{\tilde{\theta}, \phi} \arg\max_{\lambda \ge 0} \left(\hat{\mathfrak{L}}(\tilde{\theta}, \phi; \{\lambda\}) \right). \tag{6}$$

2.2 Entropy Maximization of Stochastic Gates

We use entropy $(H(S) = -\sum_{S \in \Omega} P(S) \log P(S))$ maximization of states $S \in \Omega = \{0,1\}^p$ given a set of constraints to derive the Boltzmann distribution and free energy. Exploiting the connection between free energy upper bound and evidence lower bound (ELBO) (Altosaar et al., 2019), we show that the constrained optimization problem that Gallego-Posada et al. (2022) propose for controlled sparsity naturally arises from the minimization of free energy upper bound with a constraint on the density of micro states. The exploration of such a connection between statistical mechanics and machine learning is not new. LeCun et al. (2006) introduce the energy-based models, where the energy corresponds to the loss of a model. Carbone (2025) review theoretical and practical aspects of energy-based models, connecting elements of statistical physics and machine learning. Lairez (2023) provide a short introduction to the derivation of boltzmann distribution and free energy.

We assume P(S) drives the state of parameters $\theta \in \mathbb{R}^p$ being non-zero with a parameterization of $\theta = \tilde{\theta} \odot S$ ($\tilde{\theta}(\neq 0)$) and thus refer to S as gates. The Lagrangian associated with entropy maximization of S, subject to normalization constraint for P(S), expected gate density constraint, and a finite constraint on the normalized loss of a reparameterized model on data D:(X,Y) where $X \in \mathbb{R}^{N \times p}$ and $Y \in \mathbb{R}^N$, can be expressed as

$$\mathfrak{L}(P(S); \{\lambda_i\}) = \sum_{\Omega} P(S) \log P(S) + \lambda_0 \left(\sum_{\Omega} P(S) - 1 \right)
+ \lambda_1 \left(\sum_{\Omega} P(S) \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} - \rho \right) + \lambda_2 \sum_{\Omega} \left(P(S) \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot S), y_i \right) \right] - L^* \right).$$
(7)

Here, the Lagrange multiplier λ_2 identified as inverse temperature $\beta=1/T$ is positive, λ_1 is nonnegative, and L^* is a finite constant. A detailed derivation of the following steps is provided in Appendix A.

The probability distribution P(S) can be expressed in terms of the Hamiltonian (H(S)) or energy function in S and the normalizing constant Z using the stationarity condition (Lairez, 2023, eq 17):

$$H(S) = \frac{1}{N} \sum_{i=1}^{N} \ell(h(x_i; \tilde{\theta} \odot S), y_i) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|}, \quad P(S) = \frac{e^{-\frac{1}{T}H(S)}}{Z}$$

$$Z = \sum_{\Omega} e^{-\frac{1}{T}H(S)} \quad \text{, where } e^{-\frac{1}{T}H(S)} \text{ is the Boltzmann factor.}$$
(8)

This distribution is intractable because the normalizing constant is not factorizable in S, resulting in combinatorial complexity. If a simpler $H_0(S)$ such as $\sum_j \lambda s_j h_j/|\theta|$ that factorizes over S is chosen as a trial Hamiltonian, then the associated trial distribution q(S) is a tractable mean field approximation of P(S) and the marginal $q(s_j)$ is a Bernoulli distribution. Using the Bogoliubov variational principle, an upper bound on the free energy can be derived (Kuzemsky, 2015, ch 8). Altosaar et al. (2019) show that minimization of the free energy upper bound is the same as minimization of negative ELBO in the Bayesian variational principle, using the Boltzmann factor as an unnormalized posterior and the trial distribution as the approximate posterior in ELBO. A lower bound on the negative ELBO (\mathcal{F}_{LB}) can further be obtained using the positivity of Kullback-Leibler divergence $\mathcal{D}_{KL}\left(q(S)\mid p(S)\right)$ between the approximate posterior and the prior of the same form. \mathcal{F}_{LB} contains expectations of the normalized loss of the reparameterized model and the gate density, with respect to q(S). The minimization of \mathcal{F}_{LB} involves discrete sampling of the gates from

q(S). Since the gradients of the model with respect to parameters of the variational distribution q(S) do not flow through the discrete sampling, a stochastic minimization procedure with Monte-Carlo estimation (Carbone, 2025, eq.27) can be employed, leading to:

$$\hat{\mathcal{F}}_{LB} = \sum_{r=1}^{R} \frac{1}{R} \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot S^{(r)}), y_i \right) \right] + \lambda E_{q(S)} \left[\sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} \right]$$
(9)

The expectation of $\sum_j s_j$ is the sum of the probabilities of non-zero gates, a differentiable relaxation of L_0 pseudo-norm counting non-zero parameters given $\tilde{\theta} \neq 0$. Choosing a continuous approximation of the Bernoulli distribution and sampling, that can be expressed as a deterministic transformation of a parameter-free noise such as hard concrete distribution, the Lagrangian for the L_0 constrained minimization of $\hat{\mathcal{F}}_{LB}$ is exactly the same as equation 5.

2.3 FL WITH REPARAMETERIZATION

In an FL setting with C clients holding data $(D^{(c)})_{c=1}^C: (X^c, Y^c)$, a reparameterized linear model $h(x; \tilde{\theta} \odot z): \mathbb{R}^p \to \mathbb{R}$, and loss $\ell(h(x; \tilde{\theta} \odot z), y)$ where $X^c \in \mathbb{R}^{n_c \times p}, Y^c \in \mathbb{R}^{n_c}, \sum_{c=1}^C n_c = N, x \in \mathbb{R}^p, y \in \mathbb{R}, \theta \in \mathbb{R}^p$ ($\theta = \tilde{\theta} \odot z$), and gate parameters $\phi = \log \alpha \in \mathbb{R}^p$, the Lagrangian for the L_0 density constrained minimization problem can be written as

$$\hat{\mathfrak{L}}(\tilde{\theta}, \phi; \lambda) = \sum_{c=1}^{C} \frac{n_c}{N} \mathcal{L}^{(c)}(\tilde{\theta}, \phi) + \lambda \left(\sum_{j=1}^{|\theta|} \frac{E_{q(z|\phi)}[z_j]}{|\theta|} - \rho \right). \tag{10}$$

Here, $\mathcal{L}^{(c)}(\tilde{\theta},\phi)$ is Monte Carlo estimate of the normalized loss at client c evaluated as

$$\mathcal{L}^{(c)}(\tilde{\theta}, \phi) = \frac{1}{R} \sum_{r=1}^{R} \frac{1}{n_c} \sum_{i=1}^{n_c} \ell\left(h(x_i^c; \tilde{\theta} \odot z^{(r)}), x_i^c)\right). \tag{11}$$

A hard concrete distribution $g(f(\phi, \epsilon))$ for sampling of gates(z) is a hard-sigmoid transformation of the stretched \bar{s} of the binary concrete random variable s. Louizos et al. (2017) proposed hard concrete as a closer approximation, allowing for zeros in z, of Bernoulli than the binary concrete (Maddison et al., 2016).

Concrete:
$$s = q(s|\phi) = \sigma\left(\frac{\log\frac{u}{1-u} + \log\alpha}{\beta'}\right), \quad u \sim \mathcal{U}(0,1),$$

Stretch: $\bar{s} = s(\zeta - \gamma) + \gamma, \quad \text{Transform: } z = \min(1, \max(0, \bar{s})).$ (12)

Using the cumulative distribution $Q(\bar{s})$, presented at Louizos et al. (2017),the min-max problem can be expressed as:

$$\tilde{\theta}^*, \phi^*, \lambda^* = \arg\min_{\tilde{\theta}, \phi} \arg\max_{\lambda \ge 0} \left(\sum_{c=1}^C \frac{n_c}{N} \mathcal{L}^{(c)}(\tilde{\theta}, \phi) + \lambda \left(\sum_{j=1}^{|\theta|} \frac{E_{q(z|\phi)}[z_j]}{|\theta|} - \rho \right) \right)$$
 where
$$E_{q(z|\phi)}[z_j] = 1 - Q(\bar{s}_j \le 0|\phi_j) = \sigma \left(\log \alpha_j - \beta' \log \left(-\frac{\gamma}{\zeta} \right) \right).$$
 (13)

We can now perform a joint optimization of $\tilde{\theta}$ and $\phi = \log \alpha$ using gradient descent with reparameterized gradients, and a test-time \hat{z}^r without noise and smoothing (Louizos et al., 2017). We use a gradient ascent updating rule for λ with restart (λ =0) when the constraint is satisfied (Gallego-Posada et al., 2022).

3 DISTRIBUTED OPTIMIZATION ALGORITHM

We now introduce a short notation $\mathcal{L}_{Con}(\phi)$ representing the L_0 density constraint in equation 10. The Lagrangian in short notation can be expressed as

$$\hat{\mathfrak{L}}(\tilde{\theta}, \phi; \{\lambda\}) = \sum_{c=1}^{C} \frac{n_c}{N} \mathcal{L}^{(c)}(\tilde{\theta}, \phi) + \lambda \mathcal{L}_{Con}(\phi).$$
 (14)

McMahan et al. (2017) propose a distributed algorithm for learning a global model with synchronous update using averaging of gradients. Assuming a central server that coordinates training with C clients holding data $(D^c)_{c=1}^C$ locally, clients compute the gradients and the server aggregates these gradients from the clients to update the global model. For all data at each client, clients compute $\nabla_{\tilde{\theta}} \mathcal{L}^{(c)}(\tilde{\theta},\phi)$ and $\nabla_{\phi} \mathcal{L}^{(c)}(\tilde{\theta},\phi)$ at time t and the server updates: $\tilde{\theta}^{t+1} \leftarrow \tilde{\theta}^t - \eta_{\tilde{\theta}} \sum_{c=1}^C \frac{n_c}{N} \nabla_{\tilde{\theta}} \mathcal{L}^{(c)}(\tilde{\theta}^t,\phi^t)$, $\phi^{t+1} \leftarrow \phi^t - \eta_{\phi}(\sum_{c=1}^C \frac{n_c}{N} \nabla_{\phi} \mathcal{L}^{(c)}(\tilde{\theta}^t,\phi^t) - \lambda \nabla_{\phi} \mathcal{L}_{\text{Con}}(\phi^t))$ and $\lambda^{t+1} \leftarrow \lambda^t + \mathcal{L}_{\text{Con}}(\phi^t)$.

We use mini-batches of uniform size B drawn randomly at all clients for a synchronous update of global parameters, and repeat this process for n(B) iterations in each epoch/round. A fraction γ_c of clients C is chosen at random for training in each epoch $(K = \lfloor \gamma_c \cdot C \rfloor)$. For a mini-batch at each client, clients compute $\nabla_{\tilde{\theta}} \mathcal{L}_B^{(k)}(\tilde{\theta},\phi)$ and $\nabla_{\phi} \mathcal{L}_B^{(k)}(\tilde{\theta},\phi)$, the gradients of mini-batch normalized loss with respect to $\tilde{\theta}$ and ϕ , and the server updates: $\tilde{\theta}^{t+1} \leftarrow \tilde{\theta}^t - \eta_{\tilde{\theta}} \sum_{k=1}^K w_k \nabla_{\tilde{\theta}} \mathcal{L}_B^{(k)}(\tilde{\theta}^t,\phi^t)$, $\phi^{t+1} \leftarrow \phi^t - \eta_{\phi}(\sum_{k=1}^K w_k \nabla_{\phi} \mathcal{L}_B^{(k)}(\tilde{\theta}^t,\phi^t) - \lambda \nabla_{\phi} \mathcal{L}_{\text{Con}}(\phi^t))$ and same update for λ as before but with a restart mechanism if the constraint is satisfied (Gallego-Posada et al., 2022). The weights for aggregation in our case are $w_k = \frac{B}{K \cdot B}$. With this approach, we learn a global sparse model in FL with an L_0 constraint using probabilistic gates, which we refer to as FLoPS.

The learning rates $\eta_{\tilde{\theta}} \sim [1e^{-3}, 1e^{-1}]$, $\eta_{\phi} \sim [1e^{-3}, 1e^{-1}]$, and η_{λ} in the order of $1/|\theta|$ are tuned for stability in training. The gate parameters $\log \alpha$ are sampled from normal distribution with mean of $\log \rho_{\text{init}} - \log(1-\rho_{\text{init}})$ and variance 0.01 and $\rho_{\text{targ}} = \rho$ is the desired density of parameters. A high ρ_{init} implies a dense initialization. The hyperparameters of hard concrete: γ , ζ , and β' are set at -0.1, 1.1, and 0.66 as recommended (Louizos et al., 2017). The Lagrange parameter λ is set to 0 initially.

Algorithm 1 FLOPS. E is the number of epochs and B is the mini-batch size.

```
Initialize: (\tilde{\theta}^{(0)}, \phi^{(0)} : \log \alpha^{(0)})
for epoch t = 1, \dots, E do
      S_t=random set of K clients (K = |\gamma_c \cdot C|)
      for batch b = 1, \ldots, n(B) do
           for each participating client k \in S_t do
                 gradients with respect to (\tilde{\theta}, \phi): ClientCompute(k, \tilde{\theta}^t, \phi^t)
           Aggregate gradients from clients w.r.t (\tilde{\theta}, \phi): \sum_{k=1}^{K} w_k g_{\tilde{\theta}}^k and \sum_{k=1}^{K} w_k g_{\phi}^k
           Compute gradients \nabla_{\phi} \mathcal{L}_{Con}(\phi^{(t)})
           Update (\tilde{\theta}^{(t+1)}, \phi^{(t+1)}) and the dual \lambda^{(t+1)}
     end for
     If epoch > prune start:
      scale up \log \alpha^{(b)} at top-m indices of \theta (m = \lfloor (\rho_{targ} \cdot |\theta|) \rfloor) and scale down for the remaining.
      ((\log \alpha^{(t)} + r \log \alpha^{(t)}) m_{\theta} + (\log \alpha^{(t)} - r \log \alpha^{(t)}) (1 - m_{\theta}) where r and m_{\theta}
      are decay factor and top-m mask respectively.)
end for
ClientCompute(k, \hat{\theta}, \phi): // Run for each client k \in S_t
   Draw a mini batch b \leftarrow random(D^{(k)}, B) of size B
   Compute g_{\tilde{\theta}}^k = \nabla_{\tilde{\theta}} \mathcal{L}_B^{(k)}(\tilde{\theta}^t, \phi^t) and g_{\phi}^k = \nabla_{\phi} \mathcal{L}_B^{(k)}(\tilde{\theta}^t, \phi^t)
    Send g_{\tilde{\theta}}^k and g_{\phi}^k to server
```

By design, FLOPS gives test time sparsity with \hat{z} sampled without noise and smoothing. Gallego-Posada et al. (2022) note that their approach results in a density closer to the targeted density. For exact sparsity, while not breaking the assumption of $\tilde{\theta} \neq 0$, we scale up $\log \alpha$ at top-m indices where $m = \lfloor \rho_{\text{targ}} \cdot |\theta| \rfloor$ and scale down for the rest of the indices in each epoch after a set threshold called prune start. In an FL setting, since the data available at a client varies, applying a data heterogeneity-aware method, such as the Reimannian aggregation scheme (Ahmad et al., 2023) or simply choosing $w_k = \frac{n_k}{N}$, can be tested in the future. The fact that not all clients may qualify to

participate in training is simulated by randomly choosing a fraction of clients for training in each epoch.

4 EXPERIMENTS

The experiments conducted span both synthetic and real datasets. The sparse linear regression (LR), logistic regression (LGR), and multiclass classification (MC) results on synthetically generated data are included. The experimental results of LR, MC, and multi-label classification (MLC) on real datasets e2006-tfidf, MNIST, and RCV1, respectively, are also included. We conducted all our experiments on an Apple MacBook with an M4 Pro chip (12-core CPU) and 24 GB of unified memory running macOS 15.5, using PyTorch (2.7.0) in Python (3.12.7).

4.1 EXPERIMENTS ON SYNTHETIC DATA SETS

For generating synthetic linear regression data, we use a sparse linear model following the method described at Bertsimas et al. (2020). Each row $x_i \in \mathbb{R}^{1000}$ of the design matrix $X \in \mathbb{R}^{10000 \times 1000}$ is drawn from a zero-mean Gaussian with a covariance matrix Σ . We use a Toeplitz covariance matrix Σ where $(\Sigma_{ij})_{i,j=1}^{1000} = \rho_{cor}^{|i-j|}$. An m-sparse $(m = \lfloor \rho \cdot 1000 \rfloor)$ coefficient vector $w_{\text{true}} \in \mathbb{R}^{1000}$ is constructed by randomly choosing a set of m indices $(S_m \subseteq \{1,..1000\})$ and sampling $(w_{\text{true}})_j \sim \text{Unif}\{-1,1\}$ for $j \in S_m$ and $(w_{\text{true}})_j = 0$ otherwise. This is the true sparsity of the model. The responses or predictions are generated as $y = Xw_{\text{true}} + \varepsilon$ with i.i.d. noise $\varepsilon \sim \mathcal{N}(0,\sigma^2 I_N)$ where N = 10000. A signal-to-noise ratio defined by $SNR = \|Xw_{\text{true}}\|_2^2/\|\varepsilon\|_2^2$ is choosen and the noise level $\sigma = \|Xw_{\text{true}}\|_2/(\sqrt{SNR}\sqrt{N})$ is chosen accordingly. For generating synthetic sparse logistic regression, the same procedure is used except the binary labels are obtained by thresholding noisy logits $(n_l = Xw_{\text{true}} + \varepsilon)$ by 0, i.e., $y_i = 1$ if $(n_l)_i > 0$ and zero otherwise. In the case of multiclass classification $(n_c$ classes), an m-sparse $(m = \lfloor \rho \cdot 1000 \cdot n_c \rfloor)$ coefficient matrix is constructed by randomly choosing m positions in the matrix to populate using samples from Unif $\{-1,1\}$ and zero otherwise. The labels are generated by taking the index of the largest among the noisy logits across classes $(y_i = \arg\max_{\varepsilon} \{((n_l)_i)_{\varepsilon}\})$.

We employ affine shifting and the Dirichlet partition protocol for simulating data heterogeneity, and randomly sample a fraction of clients in each epoch for client participation heterogeneity in our experiments (Solans et al., 2024; Reisizadeh et al., 2020). We refer readers to Appendix A for more details. The True Discovery Rate (TDR) (Bertsimas et al., 2020) is used as a metric, along with mean squared error (MSE) and \mathbb{R}^2 for linear regression, and cross-entropy (CE) loss and accuracy for classification, to compare the accuracy of sparsity recovery and statistical performance. We compare our results with the federated iterative hard thresholding algorithm (FedIter-HT) proposed by Tong et al. (2022), where a hard thresholding operator external to federated averaging with gradient descent is used to iteratively impose top-m magnitude selection to minimize L_0 regularized objective, showing promising results in their experiments. We tested the sparsity recovery and performance of all algorithms with varying correlation factor ρ_{cor} and SNR in synthetic data, ranging from high to low. The results for the case with low ρ_{cor} and high SNR for LR, LG, and MC at true sparsity of 5% are illustrated in Table 1. In all three cases, FLoPS performs better than FedIter-HT, and 5% dense FLOPS performs better than 95% dense FLOPS. Since we observed similar comparative results, the sparsity recovery results in other conditions are attached in Appendix A to avoid redundancy.

We conducted experiments in homogeneous (IID) and heterogeneous (Non-IID) data distribution with a 10% of client participation in both settings, controlled by the Dirichlet parameter $\alpha_{\rm iid}$, at different levels of true sparsity, specifically 5%: $\rho_{\rm targ}=0.95$ and 95%: $\rho_{\rm targ}=0.05$. Here $\rho_{\rm targ}$ is the desired density of the model. The Table 2 illustrates that <code>FLoPS</code> has superior statistical performance ($R^2/{\rm Accuracy(ACC)}$), and the sparsity recovery accuracy (TDR) consistently, especially at learning models with very low density of parameters or high sparsity.

The desired property of gradually achieving the target density of parameters during training time can be observed through a reduction in the expected number of gates, which is a continuous approximation of the L_0 pseudo-norm in FLOPS, over the training epochs. The Figure 1 demonstrates controlled sparsity learning for a target density of 0.05.

	Density (ρ_{true})	TDR			Epoch 50: Statistical performance (test time)					
Model		FLoPS(0.05)	05) FLoPS(0.95) FedIter-H		FLoPS(0.05)		FLoPS(0.95)		FedIter-HT	
					R ² /ACC	MSE/CE	R ² /ACC	MSE/CE	R ² /ACC	MSE/CE
LR	0.05	1.00	0.05	0.18	0.91	4.62	0.85	7.65	0.16	42.57
LG	0.05	0.94	0.11	0.11	0.90	0.32	0.83	0.56	0.63	0.88
MC	0.05	0.99	0.38	0.51	0.68	0.82	0.52	2.25	0.24	2.24

Table 1: This table corresponds to synthetic data generated using a signal-to-noise (SNR) ratio of 20 and a covariance matrix generated using a 0.2 correlation factor with a true model density of 0.05. It provides a comparison across models learned under data and client participation heterogeneity. On the left: TDR for FLoPS trained to achieve densities 0.05 and 0.95, and FedIter-HT trained to achieve true density; On the right: Statistical performance at test time for all models showing R^2 and MSE for LR, and Accuracy (ACC) and CE loss for classification models after training for 50 epochs.

		$\alpha_{\rm iid} = 1000 ({\rm IID})$				$\alpha_{\mathrm{iid}} = 0.5 (\mathrm{non\text{-}IID})$			
Model	Density ($\rho_{\text{targ}} = \rho_{\text{targ}}$)	FLoPS		FedIterHT		FLoPS		FedIterHT	
		R ² /ACC	TDR	R ² /ACC	TDR	R ² /ACC	TDR	R ² /ACC	TDR
LR	0.95	0.83	0.98	0.86	0.97	0.69	0.96	0.74	0.96
	0.05	0.90	1.00	0.37	0.37	0.91	1.00	0.27	0.18
LG	0.95	0.87	0.96	0.87	0.96	0.85	0.95	0.81	0.96
	0.05	0.89	0.96	0.70	0.22	0.90	0.94	0.65	0.11
MC	0.95	0.52	1.00	0.53	1.00	0.50	1	0.50	1.00
	0.05	0.71	0.99	0.28	0.51	0.68	0.99	0.24	0.5

Table 2: Comparison of FLOPS and FedIter-HT across models and densities (true model sparsity). Here, the sub-columns R^2/ACC and TDR(true discovery rate) showcase the statistical performance and accuracy in sparsity recovery, with client participation heterogeneity: 10% of clients participate in training.

The dynamic sparsity learning in FLOPS can be understood from the change in the sparsity pattern through soft Jaccard loss/ soft intersection over union (IOU) heat map (Wang et al., 2024) of test time gates across epochs, in contrast to the IOU heat map of a binary mask in FedIter-HT where a lower value indicates higher mobility in learning sparsity. The Figure 2b shows a stable, low learning phase in the beginning, starting from a dense initialization, followed by an active learning phase and a stable sparsity pattern towards the end in FLOPS as opposed to continuous change in IOU in FedIter-HT. The heat maps correspond to learning a 0.05 target density of parameters.

4.2 EXPERIMENTS ON REAL DATASETS

We considered three publicly available datasets: (i) e2006-tfidf (Kogan et al., 2009) is a regression dataset for predicting stock return volatility with tfidf vectors as features. (ii) RCV1 (Lewis et al., 2004) is a multi-label classification dataset with a tfidf representation of Reuters newswire articles as features; we used 34 labels that account for $\sim 87\%$ of all label assignments for the multi-label classification experiment. (iii) MNIST (LeCun, 1998) is a multi-class classification dataset of handwritten digits with grayscale image pixel values as features.

For conducting FL experiments, the data needs to be decentralized. Tong et al. (2022) use k-means clustering to group the data into 10 clusters and partition each cluster into 20 even parts for both e2006-tfidf and RCV1 data. A random selection of two clusters is used to sample one partition from each cluster to allocate to one of the 100 clients. We distribute two randomly selected cluster-partition pairs to each of the 100 clients. In this way, each client sees at most two clusters simulating heterogeneity. In the case of MNIST data, the Dirichlet partition protocol is applied to draw heterogeneous label proportions for each client, and the data is distributed accordingly to 100 clients (Solans et al., 2024). In each epoch, only 5% of the clients are randomly sampled to simulate client participation heterogeneity.

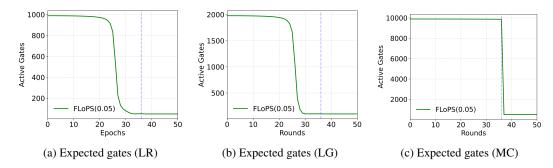


Figure 1: The figure corresponds to results for synthetic data generated using a signal-to-noise (SNR) ratio of 20 and a covariance matrix generated using a 0.2 correlation factor. Here, (a) to (c) correspond to the expected gates of FLoPS: achieving 5% target density of gates during training in heterogeneous conditions of data and client participation (HTC) in LR, LG, and MC cases, respectively. The blue dotted line corresponds to the round at which $\log \alpha$ scaling starts, corresponding to the target density.

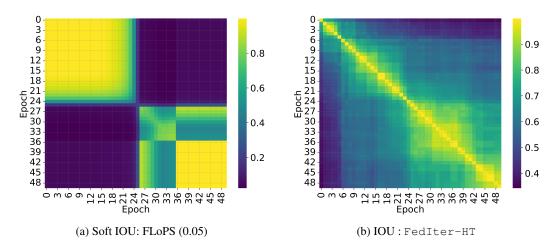


Figure 2: The figure corresponds to results for synthetic data generated using a signal-to-noise (SNR) ratio of 20 and a covariance matrix generated using a 0.2 correlation factor. Here, (a) and (b) correspond to the soft IOU heat map for test time gates in FLoPS for 5% target density of gates and IOU heat map of binary masks in FedIter-HT for the same level target density during training in heterogeneous conditions of data and client participation (HTC) in the LR case, respectively.

With 150360 and 47236 dimensions, the e2006-tfidf and RCV1 have high-dimensional and sparse features, leading to models LR with 150360 parameters and MLC with ~ 1.6 million parameters. The MNIST data has 28×28 pixel values as features, leading to MC with 7840 parameters.

The MSE- R^2 , binary CE (BCE)-micro-averaged F_1 , and CE-accuracy are used for comparing the statistical performance of LR on e2006-tfidf, MLC on RCV1, and MC on MNIST, respectively. Figure 3 shows that FLoPS (0.005) has a superior test time performance compared to FedIter-HT in the RCV1 multi-label classification task, with higher micro-averaged F1 and similar performance in e2006-tfidf volatility prediction R^2 and MNIST multi-class classification accuracy.

5 CONCLUSION

We introduced FLoPS, a federated learning algorithm for L_0 density-constrained optimization, which learns a global model with desired sparsity using a reparameterization with probabilistic gates. We showed that the L_0 regularized objective (Louizos et al., 2017) and the L_0 constrained formulation (Gallego-Posada et al., 2022) can be derived from entropy maximization of stochastic gates introduced for inducing sparsity. We reformulated the min-max problem associated with the

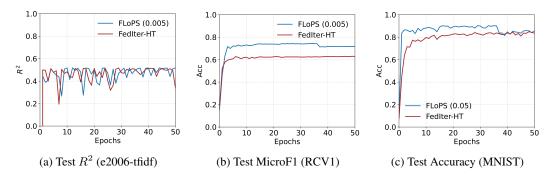


Figure 3: The figure corresponds to results of FLOPS and FedIter-HT with 0.5% target density on e2006-tfidf and RCV1 datasets, and 5% target density on MNIST data in heterogeneous conditions of data and client participation.

Lagrangian for L_0 constrained optimization for FL and proposed the distributed optimization algorithm FLOPS based on FedSGD. Through empirical evaluation using synthetically generated data and high-dimensional real data, we showed that FLOPS achieves high sparsity recovery accuracy and superior statistical performance under data and client participation heterogeneity. All the experiments currently assume a central server coordinating the synchronous training in FL. In the future, we aim to adapt the current approach to FL with decentralized coordination, eliminating the need for reliance on a central server and asynchronous updates.

REFERENCES

Adnan Ahmad, Wei Luo, and Antonio Robles-Kelly. Federated learning under statistical heterogeneity on riemannian manifolds. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 380–392. Springer, 2023.

Jaan Altosaar, Rajesh Ranganath, and Kyle Cranmer. Hierarchical variational models for statistical physics. In *Machine Learning and the Physical Sciences Workshop*, *NeurIPS* 2019, 2019. URL https://ml4physicalsciences.github.io/2019/files/NeurIPS_ML4PS_2019_92.pdf.

Dimitris Bertsimas, Jean Pauphilet, and Bart Van Parys. Sparse regression: Scalable algorithms and empirical performance. *Statistical Science*, 35(4):pp. 555–578, 2020. ISSN 08834237, 21688745. URL https://www.jstor.org/stable/26997931.

Davide Carbone. Hitchhiker's guide on the relation of energy-based models with other generative models, sampling and statistical physics: a comprehensive review, 2025. URL https://arxiv.org/abs/2406.13661.

Valeriia Cherepanova, Roman Levin, Gowthami Somepalli, Jonas Geiping, C Bayan Bruss, Andrew G Wilson, Tom Goldstein, and Micah Goldblum. A performance-driven benchmark for feature selection in tabular deep learning. Advances in Neural Information Processing Systems, 36:41956–41979, 2023.

Emanuele Frandi, Ricardo Ñanculef, Stefano Lodi, Claudio Sartori, and Johan AK Suykens. Fast and scalable lasso via stochastic frank—wolfe methods with a convergence guarantee. *Machine Learning*, 104(2):195–221, 2016.

Jose Gallego-Posada, Juan Ramirez, Akram Erraqabi, Yoshua Bengio, and Simon Lacoste-Julien. Controlled sparsity via constrained optimization or: How i learned to stop tuning penalties and love constraints. *Advances in Neural Information Processing Systems*, 35:1253–1266, 2022.

Roger A Horn. The hadamard product. In Proc. symp. appl. math, volume 40, pp. 87–169, 1990.

Iris A. M. Huijben, Wouter Kool, Max B. Paulus, and Ruud J. G. van Sloun. A review of the gumbel-max trick and its extensions for discrete stochasticity in machine learning, 2022. URL https://arxiv.org/abs/2110.01515.

- Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and trends® in machine learning*, 14(1–2):1–210, 2021.
 - Shimon Kogan, Dimitry Levin, Bryan R Routledge, Jacob S Sagi, and Noah A Smith. Predicting risk from financial reports with regression. In *Proceedings of human language technologies: the 2009 annual conference of the North American Chapter of the Association for Computational Linguistics*, pp. 272–280, 2009.
 - Alexander L Kuzemsky. Variational principle of bogoliubov and generalized mean fields in many-particle interacting systems. *International Journal of Modern Physics B*, 29(18):1530010, 2015.
 - Didier Lairez. A short derivation of boltzmann distribution and gibbs entropy formula from the fundamental postulate, 2023. URL https://arxiv.org/abs/2211.02455.
 - Yann LeCun. The mnist database of handwritten digits. http://yann. lecun. com/exdb/mnist/, 1998.
 - Yann LeCun, Sumit Chopra, Raia Hadsell, M Ranzato, Fujie Huang, et al. A tutorial on energy-based learning. *Predicting structured data*, 1(0), 2006.
 - David D Lewis, Yiming Yang, Tony G Rose, and Fan Li. Rcv1: A new benchmark collection for text categorization research. *Journal of machine learning research*, 5(Apr):361–397, 2004.
 - Christos Louizos, Max Welling, and Diederik P Kingma. Learning sparse neural networks through *l*₋0 regularization. *arXiv preprint arXiv:1712.01312*, 2017.
 - Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.
 - Brendan McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Aguera y Arcas. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pp. 1273–1282. PMLR, 2017.
 - Rajesh Ranganath. *Black Box variational inference: Scalable, generic Bayesian computation and its applications.* PhD thesis, Princeton University, 2017.
 - Amirhossein Reisizadeh, Farzan Farnia, Ramtin Pedarsani, and Ali Jadbabaie. Robust federated learning: The case of affine distribution shifts. *Advances in neural information processing systems*, 33:21554–21565, 2020.
 - Kenan Šehić, Alexandre Gramfort, Joseph Salmon, and Luigi Nardi. Lassobench: A high-dimensional hyperparameter optimization benchmark suite for lasso. In *International Conference on Automated Machine Learning*, pp. 2–1. PMLR, 2022.
 - David Solans, Mikko Heikkila, Andrea Vitaletti, Nicolas Kourtellis, Aris Anagnostopoulos, Ioannis Chatzigiannakis, et al. Non-iid data in federated learning: A survey with taxonomy, metrics, methods, frameworks and future directions. *arXiv preprint arXiv:2411.12377*, 2024.
 - Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 58(1):267–288, 1996.
 - Qianqian Tong, Guannan Liang, Jiahao Ding, Tan Zhu, Miao Pan, and Jinbo Bi. Federated optimization of ℓ 0-norm regularized sparse learning. *Algorithms*, 15(9):319, 2022.
 - Jianyu Wang, Zachary Charles, Zheng Xu, Gauri Joshi, H. Brendan McMahan, Blaise Aguera y Arcas, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, Suhas Diggavi, Hubert Eichner, Advait Gadhikar, Zachary Garrett, Antonious M. Girgis, Filip Hanzely, Andrew Hard, Chaoyang He, Samuel Horvath, Zhouyuan Huo, Alex Ingerman, Martin Jaggi, Tara Javidi, Peter Kairouz, Satyen Kale, Sai Praneeth Karimireddy, Jakub Konecny, Sanmi Koyejo, Tian Li, Luyang Liu, Mehryar Mohri, Hang Qi, Sashank J. Reddi, Peter Richtarik, Karan Singhal, Virginia Smith, Mahdi Soltanolkotabi, Weikang Song, Ananda Theertha Suresh, Sebastian U. Stich, Ameet Talwalkar, Hongyi Wang, Blake Woodworth,

Shanshan Wu, Felix X. Yu, Honglin Yuan, Manzil Zaheer, Mi Zhang, Tong Zhang, Chunxiang Zheng, Chen Zhu, and Wennan Zhu. A field guide to federated optimization, 2021. URL https://arxiv.org/abs/2107.06917.

Zifu Wang, Xuefei Ning, and Matthew B. Blaschko. Jaccard metric losses: Optimizing the jaccard index with soft labels, 2024. URL https://arxiv.org/abs/2302.05666.

A APPENDIX

DERIVATION OF OPTIMAL DISTRIBUTION P(S)

We start with the Lagrangian for maximizing entropy of the states S or gates with a normalizing constraint, constraint on expectation of density of states, and a constraint on the loss $\ell(h(x; \tilde{\theta} \odot \phi), y)$ of the reparameterized model $(h(x; \tilde{\theta} \odot S) : \mathbb{R}^p \to \mathbb{R})$ normalized over data D : (X, Y) where $x \in \mathbb{R}^p$, $y \in \mathbb{R}$, $X \in \mathbb{R}^{N \times p}$, $Y \in \mathbb{R}^N$ and $\theta \in \mathbb{R}^p(\theta = \tilde{\theta} \odot S)$. The Lagrange multiplier λ_2 , identified as inverse temperature $\beta = 1/T$, is positive, λ_1 is non-negative, and L^* is a finite constant.

$$\mathfrak{L}(P(S); \{\lambda_i\}) = \sum_{\Omega} P(S) \log P(S) + \lambda_0 \left(\sum_{\Omega} P(S) - 1 \right)
+ \lambda_1 \left(\sum_{\Omega} P(S) \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} - \rho \right) + \lambda_2 \sum_{\Omega} \left(P(S) \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot S), y_i \right) \right] - L^* \right).$$
(15)

Taking the functional derivative w.r.t. P(S) to zero, a stationarity condition, and solving for p(S) gives the Gibbs-Boltzmann distribution.

$$\frac{\partial \mathfrak{L}(P(S); \{\lambda_i\})}{\partial P(S)} = \frac{1}{N} \sum_{i=1}^{N} \ell\left(h(x_i; \tilde{\theta} \odot S), y_i\right) + T(1 + \log P(S)) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} + \mu = 0.$$

Here, $\mu = \lambda_0 T$ and $\lambda = \lambda_1 T$. Solving for $\log P(S)$:

$$\log P(S) = -\frac{1}{T} \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot S), y_i \right) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} + \mu + T \right]$$

$$P(S) = \frac{1}{Z} \exp\left(-\frac{1}{T} \left[\frac{1}{N} \sum_{i=1}^{N} \ell\left(h(x_i; \tilde{\theta} \odot S), y_i\right) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} \right] \right)$$
$$= \frac{1}{Z} \exp\left(-\frac{1}{T} H(S)\right). \tag{16}$$

Equation 16 is the Gibbs-Boltzmann distribution over states for known parameters and data. Here Z is the normalization constant and H(S) is the Hamiltonian or the energy function given by equation 17 and equation 18:

$$Z = \sum_{\Omega} \exp\left(-\frac{1}{T} \left[\frac{1}{N} \sum_{i=1}^{N} \ell\left(h(x_i; \tilde{\theta} \odot S), y_i\right) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} \right] \right). \tag{17}$$

$$H(S) = \frac{1}{N} \sum_{i=1}^{N} \ell(h(x_i; \tilde{\theta} \odot S), y_i) + \lambda \sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|}$$

$$\tag{18}$$

GIBBS-BOGOLIUBOV INEQUALITY AND ELBO

The Helmholtz free energy functional F is defined using logarithmic transformation of the partition function Z and inverse temperature β . The minimization of free energy yields the probability distribution with maximum entropy.

$$F = -\frac{1}{\beta}\log(Z).$$

The partition function Z associated with P(S) in equation 17 is intractable. A distribution q(S) with a partition function fully factorizable in states $s_i \ \forall i \in \{1, \dots |\theta|\}$ can be assumed as a trial distribution.

$$q(S) = \frac{1}{Z_0} \exp\left(-\frac{1}{T}H_0(S)\right); \quad F_0 = -\frac{1}{\beta}\log(Z_0)$$

Using the Gibbs-Bogoliubov inequality

$$F \le F_0 + \mathbb{E}_{Q(S)}[H(S) - H_0(S)] \tag{19}$$

and the relation between free energy and the entropy

$$F_0 = \mathbb{E}_{Q(S)}[H_0(S)] - T\mathcal{H}[Q], \tag{20}$$

an upper bound on true free energy can be obtained (Kuzemsky, 2015, eq 22, eq 122). This is the variational free energy upper bound expressed in

$$F \le \mathbb{E}_{Q(S)}[H(S)] - T\mathcal{H}[Q]. \tag{21}$$

Altosaar et al. (2019) show the relation between the Gibbs-Bogoliubov inequality for a system with null data using a mean field (MF) trial distribution and the evidence lower bound (ELBO), where the unnormalized posterior distribution corresponds to the Boltzmann factor $(e^{-\frac{1}{T}E})$ in Gibbs-Boltzmann distribution with energy function E. The ELBO can be expressed using the log joint or the unnormalized posterior corresponding to the Gibbs-Boltzmann factor in our context:

$$\begin{split} \mathbb{L} &= \mathbb{E}_{q(S)}[\log P(D,S)] - \mathbb{E}_{q(S)}[\log q(S)] \\ &= \mathbb{E}_{q(S)}[\log P(D|S)] - \mathbb{E}_{q(S)}[\log q(S)] + \mathbb{E}_{q(S)}[\log p(S)] \\ &= \mathbb{E}_{q(S)}[\log P(D|S)] - \mathcal{D}_{KL}[q(S)|p(S)]. \end{split}$$

The minimization of negative ELBO or variational free energy is the same as the minimization of the upper bound on the free energy described in equation equation 21 (Altosaar et al., 2019):

$$F = -\mathbb{L} = \mathbb{E}_{q(S)}[-\log P(D|S)] + \mathcal{D}_{KL}[q(S) \mid p(S)]$$

$$\geq \mathbb{E}_{q(S)}[-\log P(D|S)] = \mathcal{F}_{LB}$$
(22)

MINIMIZATION OF THE BAYESIAN VARIATIONAL FREE ENERGY

The minimization of the lower bound \mathcal{F}_{LB} on the Bayesian variational free energy derived using the positivity of $\mathcal{D}_{KL}(q(S) \mid p(S))$ involves sampling the state variables from q(S). For a choice of $H_0(S) = \sum_j \lambda s_j h_j / |\theta|$, q(S) is a mean field approximation of P(S) and fully factorizable in S. The probability distribution $q(s_j)$ is a Bernoulli probability. Since the gradients do not flow through the discrete sampling from the Bernoulli distribution, a stochastic minimization procedure with Monte-Carlo estimation, although inefficient, can be employed (Carbone, 2025, eq.27):

$$\hat{\mathcal{F}}_{LB} = \sum_{r=1}^{R} \frac{1}{R} \left[-\log P(D \mid S^{(r)}) \right]$$

$$= \sum_{r=1}^{R} \frac{1}{R} \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot S^{(r)}), y_i \right) \right] + \lambda E_{q(S)} \left[\sum_{j=1}^{|\theta|} \frac{s_j}{|\theta|} \right]$$
(23)

Ranganath (2017) provide a recipe for efficiently working with stochastic optimization by employing variance reduction methods such as reparametrized gradients. However, this approach excludes the

usage of discrete latent random variables S and assumes access to the gradient of the log joint or the model with respect to latent variables. Further assuming that the sampling of continuous latent variables S or gates can be expressed as a deterministic transformation of a parameter-free noise ϵ , one can simplify the stochastic optimization at hand as joint optimization of the model parameters and the gate parameters using reparameterized gradients:

$$S^{(r)} = f(\phi, \epsilon^{(r)})$$

 $\epsilon^{(r)} \sim p(\epsilon).$

Louizos et al. (2017) propose a hard concrete distribution($g(f(\phi, \epsilon))$) for continuous sampling of z, allowing for a closer approximation of the Bernoulli distribution:

$$\hat{\mathcal{F}}_{LB} = \sum_{r=1}^{R} \frac{1}{R} \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot z^{(r)}), y_i \right) \right] + \lambda E_{q(z|\phi)} \left[\sum_{j=1}^{|\theta|} \frac{z_j}{|\theta|} \right]. \tag{24}$$

Here, z is a hard-sigmoid transformation of the stretched \bar{s} of the binary concrete random variable s. Using the cumulative distribution $Q(\bar{s})$ this expression can be expressed as (Louizos et al., 2017):

$$\hat{\mathcal{F}}_{LB} = \sum_{r=1}^{R} \frac{1}{R} \left[\frac{1}{N} \sum_{i=1}^{N} \ell \left(h(x_i; \tilde{\theta} \odot z^{(r)}), y_i \right) \right] + \lambda \left[\sum_{j=1}^{|\theta|} \frac{\sigma \left(\log \alpha_j - \beta' \log \left(-\frac{\gamma}{\zeta} \right) \right)}{|\theta|} \right], \quad (25)$$

where,
$$z^{(r)} = \min(1, \max(0, \bar{s}^{(r)}))$$
, $\bar{s}^{(r)} = s^{(r)}(\zeta - \gamma) + \gamma$, $s^{(r)} = q(s^{(r)} \mid \phi) = \sigma\left(\log\alpha^{(r)} + \log\left(\frac{u^{(r)}}{1 - u^{(r)}}\right)\right)$, $u^{(r)} \sim \mathcal{U}(0, 1)$, and $E_{q(z|\phi)}[z_j] = 1 - Q(\bar{s}_j \leq 0) = \sigma\left(\log\alpha_j - \beta'\log\left(-\frac{\gamma}{\zeta}\right)\right)$.

Learning Sparsity through test time gates \hat{z}

Figure 4 illustrates how FLOPS learns the desired sparsity, using test time gates over epochs and soft jaccard loss/IOU (Wang et al., 2024). The lower the soft IOU, the higher the change or learning in the sparsity pattern.

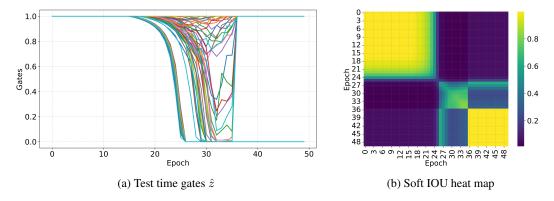


Figure 4: The figure corresponds to FLOPS (0.05) for synthetic linear regression data generated using a signal-to-noise (SNR) ratio of 20 and a Covariance matrix generated using a 0.2 correlation factor $\rho_{targ} = 0.05$. Here, (a) and (b) correspond to the test time gates \hat{z} and Soft IOU (Intersection Over Union) heat map of \hat{z} over epochs under data (non-IID) and client participation heterogeneity (10%).

MORE EXPERIMENTS ON SYNTHETIC DATA

702

703 704

705706

708

709

710

711

712 713 714

715 716

717

718

719

720 721

722

723724725

726

727

728

729

730

731

732

733

734735736

737 738

739

740

741

742 743

744

745

746

747

748

749

750

751

752

753

754

755

윤 0.6

0.4

0.2

0.0

FLOPS (0.05)

0.98 1.0 0.96 1.0 1.0 0.8 0.8 0.74 0.8 윤 0.6 원 0.6 0.6 0.6 0.4 0.4 0.4 0.20 0.2 0.2 0.2 0.12 0.05 0.05 0.0 0.0 0.0 FLOPS (0.05) FLOPS (0.95) Fediter-HT FLOPS (0.05) Fediter-HT FLOPS (0.05) FLOPS (0.95) FLOPS (0.95) (a) LR: low SNR low ρ_{cor} (b) LR: low SNR high ρ_{cor} (c) LR: high SNR high ρ_{cor} 1.0 1.0 1.0 0.89 0.8 0.8 0.8 0.59 ~ 0.6 ~ 0.6 ద్ద 0.6 0.4 0.4 0.4 0.2 0.2 0.2 0.13 0.05 0.0 0.0 FLOPS (0.05) FLOPS (0.95) FLOPS (0.05) Fediter-HT FLOPS (0.05) FLOPS (0.95) Fediter-HT FLOPS (0.95) Fediter-HT (d) LG: low SNR low ρ_{cor} (e) LG: low SNR high ρ_{cor} (f) LG: high SNR high ρ_{cor} 1.0 1.0 1.0 0.93 0.82 0.79 0.8 0.8 0.8

Figure 5: The figure corresponds to sparsity recovery results of FLoPS and FedIter-HT in data and client participation heterogeneous conditions for synthetic linear regression (LR), logistic (LG) and multiclass classification (MC) data generated in low SNR (3) - low ρ_{cor} (0.2), low SNR (3) - high ρ_{cor} (0.7), and, high SNR (20) - high ρ_{cor} (0.7).

(h) MC: low SNR high ρ_{cor}

0.38

FLOPS (0.95)

원 0.6

0.4

0.2

0.0

FLOPS (0.05)

0.48

Fediter-HT

0.38

FLOPS (0.95)

(i) MC: high SNR high ρ_{cor}

0.48

Fediter-HT

원 0.6

0.4

0.2

0.0

FLOPS (0.05)

0.51

Fediter-HT

0.38

FLOPS (0.95)

(g) MC: low SNR low ρ_{cor}

Figure A shows the sparsity recovery accuracy (TDR) of FLoPS and FedIter-HT in additional experimental conditions to those shown in the experiments section. We generated synthetic data by varying the SNR and the Toeplitz covariance matrix ($\Sigma_{ij} = \rho_{\rm cor}^{|i-j|}$) for various values of correlation factor $\rho_{\rm cor}$. We generated results in four regimes, varying SNR from 20 (high) to 3 (low) and $\rho_{\rm cor}$ from 0.2 (low) to 0.7 (high): (i) high SNR-low $\rho_{\rm cor}$ (ii) low SNR-low $\rho_{\rm cor}$ (iii) low SNR-high $\rho_{\rm cor}$ (iv) high SNR-high $\rho_{\rm cor}$. The results for the first regime are discussed in the experiments section, and the results for the remaining regimes are presented in Figure A. We see that the sparsity recovery is consistently better with FLOPS in experiments for the LR, LG, and MC tasks. We also note that the FLOPS with 95% target density is trivially poor in sparsity recovery, as we forced the higher density of parameters despite our knowledge of the inherent sparsity. Cherepanova et al. (2023) show that the statistical performance drops with an increase in corrupted or duplicated features. We suspect that the significantly poor performance of FedIter-HT in our synthetic data with dense correlated features, in contrast to real data with sparse features, could stem from the same reason and needs further investigation.

HETEROGENEITY

In FL, the data is neither centralized nor independent and identically distributed (IID). The data that each client holds is private and may not have the same quantity or attributes of data as the other clients. In practice, the same number of clients may not be eligible or available to participate throughout the training time. Solans et al. (2024) reviews various reasons for the above-mentioned forms of heterogeneity (non-IIDness) and different ways to simulate such conditions. To simulate heterogeneity in the number of samples across clients or quantity skew, a Dirichlet distribution with parameter α is used to sample proportions for each client. The samples are then allocated to clients according to the sampled proportions, using a Dirichlet partition protocol to achieve quantity skew. For a high $\alpha_{\rm iid}=1000$, the samples are uniformly distributed across 100 clients, whereas a low $\alpha_{\rm iid}=0.5$ results in a heterogeneous distribution of samples. For achieving attribute skew, affine shifts that are randomly sampled using a zero-mean Gaussian distribution with a standard deviation σ_{ms} are then used to shift features at a client by samples from a Gaussian distribution with an affine shift as mean and standard deviation of 1. Reisizadeh et al. (2020) discuss the decline in performance of the model in FL settings with attribute skew simulated using affine shifts. Finally, the client participation skew is simulated by randomly sampling 5% of the 100 clients in each round/epoch.

CONCRETE DISTRIBUTION

Maddison et al. (2016) propose the binary concrete distribution with parameters $\log \alpha_j$, using a Gumbel max trick on the Bernoulli distribution. Huijben et al. (2022) present a review of the Gumbel max trick in machine learning as a method to generate continuous samples from a deterministic transformation of an IID noise that results in the categorical probabilities. By definition, $\log \alpha_j$ are the logits of the Bernoulli probabilities π_j and can be initialized using a normal distribution with a mean of $\log(\rho/1-\rho)$ where ρ determines the expected number of non-zero parameters through the reparameterization, implying dense to sparse and sparse to sparse model training is possible.

$$\log \alpha_j^{(0)} \sim \mathcal{N}\left(\log \frac{\rho}{1-\rho}, \sigma^2\right) \quad \text{where } \rho \in (0,1)$$
 (26)