hetFL

shamsiiat.abdurakhmanova

June 18, 2024

1 Numerical experiments

1.1 Datasets

1.1.1 Synthetic Dataset

Experiments were performed on a synthetic dataset whose empirical graph $\mathcal{G}$ is partitioned into 3 equal-sized clusters $\mathcal{P} = \{\mathcal{C}^{(1)}, \mathcal{C}^{(2)}, \mathcal{C}^{(3)}\}$, with $|\mathcal{C}^{(1)}| = |\mathcal{C}^{(2)}| = |\mathcal{C}^{(3)}|$. We denote the cluster assignment of node $i \in \mathcal{V}$ by $c^{(i)} \in \{1, 2, 3\}$.

For experiments where graph connectivity is known, the edges in $\mathcal{G}$ are generated via realizations of independent binary random variables $b_{i,i'} \in \{0, 1\}$. These random variables are indexed by pairs $i, i^{'}$ of nodes that are connected by an edge $\{i, i^{'}\} \in \mathcal{E}$ if and only if $b_{i,i'} = 1$.
Two nodes in the same cluster are connected with probability $Prob\{b_{i,i'} = 1\} := p_{in}$ if nodes i, i' belong to the same cluster. In contrast, $Prob\{b_{i,i'} = 1\} := p_{out}$ if nodes $i, i^{'}$ belong to different clusters. Every edge in $\mathcal{G}$ has the same weight, $A_e = 1$ for all $e \in \mathcal{E}$.

Each node $i \in \mathcal{V}$ of the empirical graph $\mathcal{G}$ holds a local dataset $\mathcal{D}^{(i)}$ of the form $\mathcal{D}^{(i)} := \{(\mathbf{x}^{(i,1)}, y^{(i,1)}), ..., (\mathbf{x}^{(i,m_i)}, y^{(i,m_i)})\}$. Thus, dataset $\mathcal{D}^{(i)}$ consist of m_i data points, each characterized by a feature vector $\mathbf{x}^{(i,r)} \in \mathbb{R}^d$ and scalar label $y^{(i,r)}$, for $r = 1, ..., m_i$. The feature vectors $\mathbf{x}^{(i,r)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d\times d})$, are drawn i.i.d. from a standard multivariate normal distribution.

The labels of the data points are generated by a noisy linear model

$$y^{(i,r)} = (\mathbf{w}^{(i)})^T \mathbf{x}^{(i,r)} + \varepsilon^{(i,r)} \tag{1}$$

The noise $\varepsilon^{(i,r)} \sim \mathcal{N}(0, 1)$, for $i \in \mathcal{V}$ and $r = 1, .., m_i$, are i.i.d. realizations of a normal distribution. The true underlying vector $\overline{\mathbf{w}}^{(i)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d\times d})$ is drawn from a standard normal distribution and is the same for nodes from the same cluster, i.e. $\overline{\mathbf{w}}^{(i)} = \overline{\mathbf{w}}^{(i')}$ if $c^{(i)} = c^{(i')}$.

(!!! Not relevant if we only interested in weight vector est.error!!!) Datasets are divided into training and validation subsets by using resampling with replacement. The size of the validation subset is $m_i^{(val)} = 100$.

1.1.2 Shared Dataset

The dataset $\mathcal{D}^{(test)} := \{\mathbf{x}^{(i)}, ..., \mathbf{x}^{(m')}\}$ consist of only feature vectors (no labels), where feature vector is sampled as $\mathbf{x}^{(r)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_{d\times d})$. Predictions on $\mathcal{D}^{(test)}$ are shared across all nodes. The size of the dataset is $m' = 100$.

1.2 Experiments

1.2.1 Synthetic Dataset, linreg model, graph is known

In these experiments empirical graph $\mathcal{G}$ consist of $N = 150$ nodes partitioned into three clusters ($|\mathcal{C}^{(i)}| = 50$). Two nodes in the same cluster are connected with probability $p_{in} = 0.8$ if nodes $i, i^{'}$ belong to the same cluster and $p_{out} = 0.2$ if nodes $i, i^{'}$ belong to different clusters.

Each node $i \in \mathcal{V}$ of the empirical graph $\mathcal{G}$ holds a local dataset $\mathcal{D}^{(i)}$ consisting of m_i data points, each characterized by a feature vector $\mathbf{x}^{(i,r)} \in \mathbb{R}^d$ and scalar label $y^{(i,r)}$, for $r = 1, ..., m_i$, where $d = 10$. The sample size of the shared dataset $\mathcal{D}^{(test)}$ is $m' = 100$.

To learn the local parameters $\mathbf{w}^{(i)}$, we use Algorithm 1. FedRelax Least-Squares Regression with local loss

$$L_{(i)}(h^{(i)}) = \frac{1}{m_i} \sum_{r=1}^{m_i} \Big(y^{(i,r)} - h^{(i)}(\mathbf{x}^{(i,r)})\Big)^2 \tag{2}$$

and regularizer

$$\frac{\lambda}{2m'} \sum_{i' \in \mathcal{V}} A_{i,i'} \sum_{r=1}^{m'} \Big(h^{(i)}(\mathbf{x}^{(r)}) - h^{(i^{'})}(\mathbf{x}^{(r)})\Big)^2 \tag{3}$$

Algorithm 1 FedRelax Least-Squares Regression (Adjacency matrix is known)

Input: empirical graph $\mathcal{G}$ with edge weights A_{ij}; local loss $L_{(i)}(\cdot)$; shared dataset $D^{(test)} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m')}\}$; GTV parameter λ;

Initialize: $k := 0$; predictions on shared set $\{\widehat{h}_0^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}} := 0$ for all nodes $i \in \mathcal{V}$.

1: **while** stopping criterion is not met do **do**
2: **for** all nodes $i \in \mathcal{V}$ in parallel **do**
3: share predictions $\{\widehat{h}_k^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}}$, with neighbours $i' \in \mathcal{N}^{(i)}$
4: update local hypothesis $\widehat{h}_k^{(i)}$ by

$$\widehat{h}_{k+1}^{(i)} \in \arg\min_{h^{(i)}\in\mathcal{H}^{(i)}} \Bigg[\frac{1}{m_i}\sum_{r=1}^{m_i}\Big(y^{(i,r)} - h^{(i)}(\mathbf{x}^{(i,r)})\Big)^2 + \frac{\lambda}{m'}\sum_{i'\in\mathcal{N}^{(i)}} A_{i,i'}\sum_{r=1}^{m'}\Big(h^{(i)}(\mathbf{x}^{(r)}) - \widehat{h}_k^{(i')}(\mathbf{x}^{(r)})\Big)^2\Bigg]$$

5: **end for**
6: $k := k + 1$
7: **end while**

For the experiment we use linear model implemented with pytorch (no bias, optimizer SGD learning rate 0.01). We try different local dataset sizes $m_i = \{2, 3, 5, 10, 20\}$ and regularization strength $\lambda = \{0, 1, 5\}$. Tried ratio $|\mathcal{C}^{(i)}|m_i/d = \{1, 1.5, 2.5, 5, 10\}$ or $d/m_i = \{5, 3.3, 2, 1, 0.5\}$

As stopping criterion in Algorithm 1, we use a fixed number of R = 2000 iterations. For pytorch models one iteration is equivalent to one gradient step.

The results of the experiments is depicted in Figure 1 as the plot of mean estimation error (mean over 10 repetitions of the experiment for each pair of $\{\lambda, m_i\}$).

$$\frac{1}{N}\sum_{i=1}^{N}||\overline{\mathbf{w}}^{(i)} - \widehat{\mathbf{w}}^{(i)}||_2^2 \tag{4}$$

The 2nd row of the Figure 1 shows estimation error scaled by estimation error of the "pooled" model (model trained on all datasets belonging to the same cluster).

On each repetition new local $\mathcal{D}^{(i)}$ (new realizations of feature vectors and cluster-specific weights are drawn) and shared $\mathcal{D}^{(test)}$ (new realizations of feature vectors are drawn) datasets were generated. The shaded region is ± one standard deviation.

We can see from the Figure 1, that adding penalty term helps to converge to the true weight vector, the effect more pronounced for the larger d/m ratio. Large values of λ leads to sightly higher and noisier weight error estimates, but faster convergence (subplot with $\lambda = 5$).

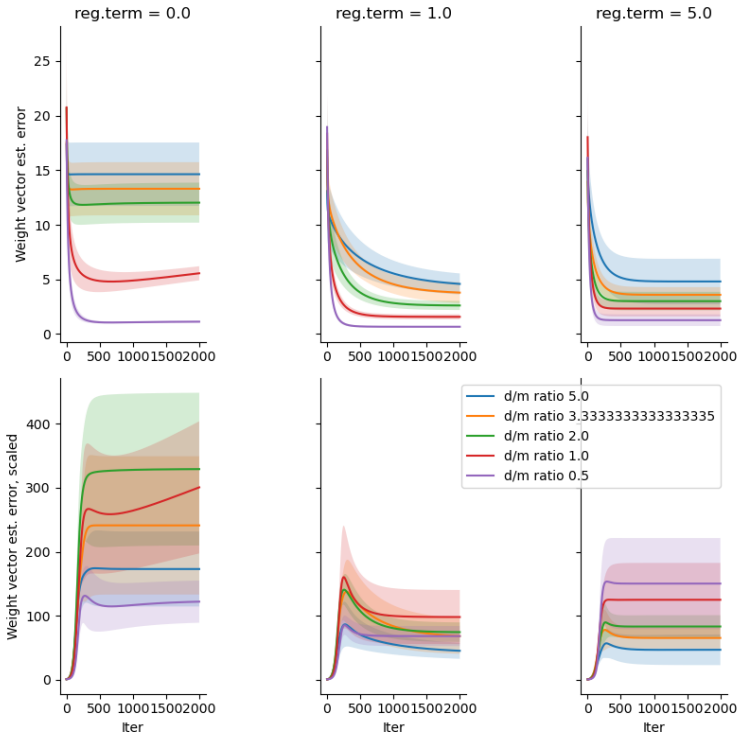

Figure 1: Algorithm 1 weight vector estimation error. Subplots corresponds to different regularization parameter values and lines correspond to different ratio d/m of a feature vector and local dataset size.

1.2.2 Sanity check

Below I do sanity check for the case where $p_{in} = 1$ and $p_{out} = 0$. The expected estimation error (eq.4) is

$$\begin{aligned} E_{est} &= \mathbb{E}\{||\overline{\mathbf{w}} - \widehat{\mathbf{w}}||_2^2\} \\ &= \mathbb{E}\{||\widehat{\mathbf{w}} - \mathbb{E}\{\widehat{\mathbf{w}}\}||_2^2\} + \mathbb{E}\{||\overline{\mathbf{w}} - \mathbb{E}\{\widehat{\mathbf{w}}\}||_2^2\} \end{aligned} \tag{5}$$

The first component represents the (expected) variance of the learnt parameter vector $\widehat{\mathbf{w}}$. The second component is referred to as a bias term. It can be shown that the variance term is

$$V = \mathbb{E}\{||\widehat{\mathbf{w}} - \mathbb{E}\{\widehat{\mathbf{w}}\}||_2^2\} = \frac{(B^2 + \sigma^2)d}{m - d - 1} \tag{6}$$

E.g. for the case $B^2 = 0, \sigma^2 = 1, d = 10, m = 20, |\mathcal{C}^{(i)}| = 50$ variance term is $10/(20 - 1 - 1) \approx 1.11$ for a local model and $10/(20 * 50 - 10 - 1) \approx 0.01$

for pooled model. For $\lambda = 0, p_{in} = 1$ and $p_{out} = 0$ the scaled estimated error is $\frac{E_{est}}{E_{est\_pooled}} = 1.11/0.01 \approx 111$. For the large values of λ the ratio is expected to be close to 1.

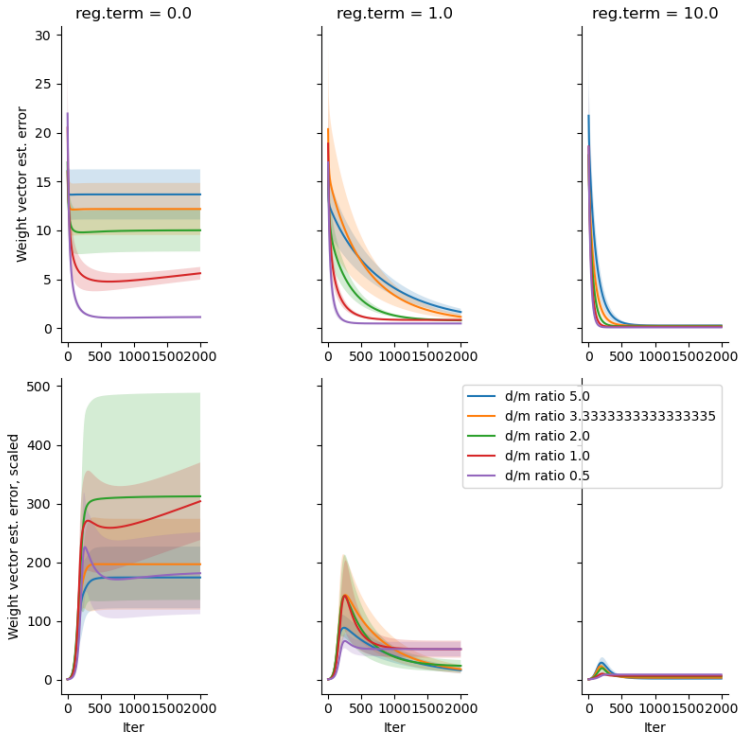

Figure 2: Algorithm 1 for params $p_{in} = 1, p_{out} = 0$

From the Figure 2 we can see that according to numerical experiments $\frac{E_{est}}{E_{est\_pooled}} \approx 180$ for the $m = 20$, which is somewhat higher than estimate ≈ 111, but of the same order. Increasing λ leads to decrease of estimation error (≈ 9 according to numerical experiments).

1.2.3 Synthetic Dataset, linreg model, graph is not known

The variation of Algorithm 1, Algorithm 2a, is used for the case when the graph connectivity is not known. In Algorithm 2a we start with adjacency matrix set to zero and on each iteration we add p links between nodes with smallest discrepancy in predictions on the shared test set $\mathcal{D}^{(test)}$. Thus, on each iteration each node's degree is either increasing or stays unchanged (but not decreasing). See resulting plots in Fig.3 and 4.

Algorithm 2 FedRelax Least-Squares Regression (Adjacency matrix is not known, degree of the node is not fixed)

Input: empirical graph $\mathcal{G}$; local loss $L_{(i)}(\cdot)$; shared dataset $D^{(test)} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m')}\}$; GTV parameter λ;

Initialize: $k := 0; A := 0$; predictions on shared set $\{\widehat{h}_0^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}} := 0$ for all nodes $i \in \mathcal{V}$.

1: **while** stopping criterion is not met do **do**
2: **for** all nodes $i \in \mathcal{V}$ in parallel **do**
3: share predictions $\{\widehat{h}_k^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}}$, with neighbours $i' \in \mathcal{N}^{(i)}$
4: update local hypothesis $\widehat{h}_k^{(i)}$ by

$$\widehat{h}_{k+1}^{(i)} \in \arg\min_{h^{(i)}\in\mathcal{H}^{(i)}} \Bigg[\frac{1}{m_i}\sum_{r=1}^{m_i}\Big(y^{(i,r)} - h^{(i)}(\mathbf{x}^{(i,r)})\Big)^2 + \frac{\lambda}{m'}\sum_{i'\in\mathcal{N}^{(i)}} A_{i,i'}\sum_{r=1}^{m'}\Big(h^{(i)}(\mathbf{x}^{(r)}) - \widehat{h}_k^{(i')}(\mathbf{x}^{(r)})\Big)^2\Bigg]$$

5: **end for**
6: **for** all nodes $i \in \mathcal{V}$ in parallel **do**
7: Add neighbours (edges) for the node i by selecting p nodes $i' \in \mathcal{V}$ with smallest values of

$$\frac{1}{m'}\sum_{r=1}^{m'}\Big(\widehat{h}_{k+1}^{(i)}(\mathbf{x}^{(r)}) - \widehat{h}_{k+1}^{(i')}(\mathbf{x}^{(r)})\Big)^2 \tag{7}$$

8: Set $A_{i,i'} = 1$ for these p nodes (with smallest discrepancy in prediction on the test set $\mathcal{D}^{(test)}$).
9: **end for**
10: $k := k + 1$
11: **end while**

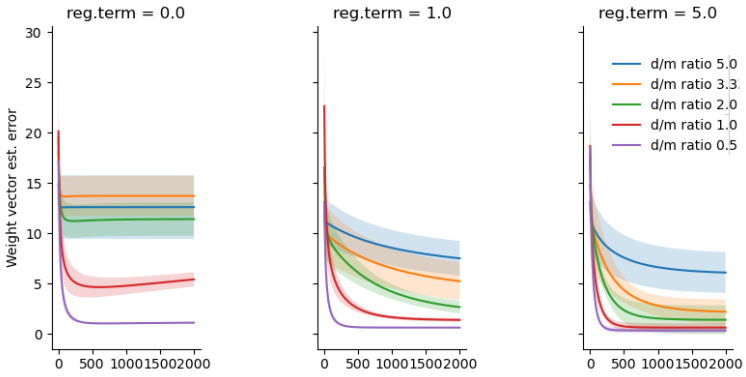

Figure 3: Algorithm 2a. Estimated error for the weight vector with parameter p=10

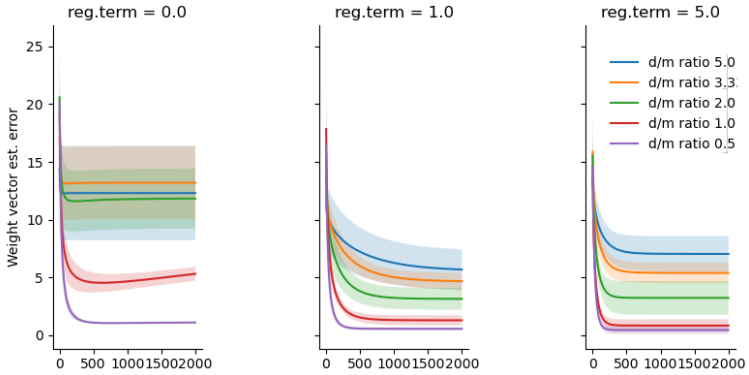

Figure 4: Algorithm 2a. Estimated error for the weight vector with parameter p=50.

In another variation, Algorithm 2b, we set adjacency matrix to zero on each iteration and then add p links between nodes with smallest discrepancy in predictions on the shared test set $\mathcal{D}^{(test)}$. Thus, the degree of a node is fixed to p. See resulting plot in Fig.5.

Algorithm 3 FedRelax Least-Squares Regression (Adjacency matrix is not known, n.o. neib. is fixed)

Input: empirical graph $\mathcal{G}$; local loss $L_{(i)}(\cdot)$; shared dataset $D^{(test)} = \{\mathbf{x}^{(1)}, ..., \mathbf{x}^{(m')}\}$; GTV parameter λ;

Initialize: $k := 0; A := 0$; predictions on shared set $\{\widehat{h}_0^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}} := 0$ for all nodes $i \in \mathcal{V}$.

1: **while** stopping criterion is not met do **do**
2: **for** all nodes $i \in \mathcal{V}$ in parallel **do**
3: share predictions $\{\widehat{h}_k^{(i)}(\mathbf{x})\}_{\mathbf{x}\in\mathcal{D}^{(test)}}$, with neighbours $i' \in \mathcal{N}^{(i)}$
4: update local hypothesis $\widehat{h}_k^{(i)}$ by

$$\widehat{h}_{k+1}^{(i)} \in \arg\min_{h^{(i)}\in\mathcal{H}^{(i)}} \Bigg[\frac{1}{m_i}\sum_{r=1}^{m_i}\Big(y^{(i,r)} - h^{(i)}(\mathbf{x}^{(i,r)})\Big)^2 + \frac{\lambda}{m'}\sum_{i'\in\mathcal{N}^{(i)}} A_{i,i'}\sum_{r=1}^{m'}\Big(h^{(i)}(\mathbf{x}^{(r)}) - \widehat{h}_k^{(i')}(\mathbf{x}^{(r)})\Big)^2\Bigg]$$

5: **end for**
6: **for** all nodes $i \in \mathcal{V}$ in parallel **do**
7: **Init a new adjacent matrix A (all zeros)**
8: Add neighbours for the node i by selecting p nodes $i' \in \mathcal{V}$ with smallest values of

$$\frac{1}{m'}\sum_{r=1}^{m'}\Big(\widehat{h}_{k+1}^{(i)}(\mathbf{x}^{(r)}) - \widehat{h}_{k+1}^{(i')}(\mathbf{x}^{(r)})\Big)^2 \tag{8}$$

9: Set $A_{i,i'} = 1$ for these p nodes (with smallest discrepancy in prediction on the test set $\mathcal{D}^{(test)}$).
10: **end for**
11: $k := k + 1$
12: **end while**

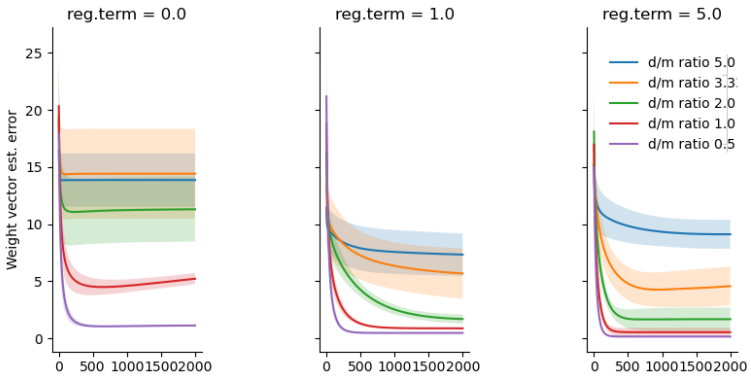

Figure 5: Algorithm 2b. Estimated error for the weight vector with parameter p=50.