Is Local SGD Better than Minibatch SGD?

A Comparative Analysis Based on Woodworth et al.

Evgeny Gurov

January, 29-th

Problem

The following stochastic convex optimization problem is considered:

$$\min_{x \in \mathbb{R}^d} F(x) := \mathbb{E}_{z \sim D}\left[f(x; z)\right]. \tag{1}$$

The focus is on objectives F that are H-smooth, either (general) convex or λ-strongly convex, with a minimizer $x^* \in \arg\min_x F(x)$ satisfying $\|x^*\| \leq B$. ∇f is assumed to have uniformly bounded variance.

Local SGD

Algorithm Local SGD Algorithm

```
1: Input: Number of workers $M$, number of communication rounds $R$, local steps $K$, learning rate $\eta$
2: Initialize parameters $\mathbf{x}_0$ (shared across all workers)
3: for $r = 0$ to $R - 1$ do
4:     for each worker $m = 1$ to $M$ in parallel do
5:         Set local parameters: $\mathbf{x}_r^m \leftarrow \mathbf{x}_r$
6:         for $k = 1$ to $K$ do
7:             Compute stochastic gradient $\mathbf{g}_k^m = \nabla f(\mathbf{x}_{k-1}^m; z_k^m)$ using local data $z_k^m$
8:             Update local parameters:
                   $\mathbf{x}_k^m = \mathbf{x}_{k-1}^m - \eta \mathbf{g}_k^m$
9:         end for
10:    end for
11:    Average the local parameters across all workers:
           $\mathbf{x}_{r+1} = \frac{1}{M} \sum_{m=1}^{M} \mathbf{x}_K^m$
12: end for
13: Output: Final parameters $\mathbf{x}_R$
```

Minibatch SGD

Algorithm Minibatch SGD

```
1:  Input: Number of workers $M$, number of rounds $R$, local steps $K$, minibatch size $KM$, learning rate $\eta$
2:  Initialize parameters $\mathbf{x}_0$ (shared across all workers)
3:  for $r = 0$ to $R - 1$ do
4:      For each worker $m = 1$ to $M$ in parallel:
5:      for $k = 1$ to $K$ do
6:          Compute the stochastic gradient:
                $\mathbf{g}_k^{(m)} = \nabla f(\mathbf{x}_{k-1}^{(m)}; z_k^{(m)})$
7:      end for
8:      Average all $KM$ gradient estimates:
                $\mathbf{g}_r = \frac{1}{KM} \sum_{m=1}^{M} \sum_{k=1}^{K} \mathbf{g}_k^{(m)}$
9:      Update parameters:
                $\mathbf{x}_{r+1} = \mathbf{x}_r - \eta \mathbf{g}_r$
10: end for
11: Output: Final parameters $\mathbf{x}_R$
```

How to compare algorithms?

Comparison of the algorithms is performed based on worst-case performance with respect to $F(H, \lambda, B, \sigma^2)$:

$$\epsilon_A = \max_{(f,D)\in F(H,\lambda,B,\sigma^2)} (F(\hat{x}_A) - F(x^*))$$

The good news is that worst-case performance of Minibatch SGD for general convex objectives is tightly understood:

$$\epsilon_{\text{MB-SGD}} = \Theta\left(\frac{HB^2}{R} + \frac{\sigma B}{\sqrt{MKR}}\right).$$

Quadratic objectives

For any quadratic $(f, D) \in F(H, \lambda = 0, B, \sigma^2)$, there is a constant c such that local-SGD returns a point $\hat{x}$ such that

$$\mathbb{E}F(\hat{x}) - F^* \leq c\left(\frac{HB^2}{KR} + \frac{\sigma B}{\sqrt{MKR}}\right),$$

which is significantly better than Minibatch SGD guarantee

$$\varepsilon_{\text{MB-SGD}} = \Theta\left(\frac{HB^2}{R} + \frac{\sigma B}{\sqrt{MKR}}\right).$$

General convex case

In the general convex case authors present the upper bound for the worst-case performance of Local SGD. When natural scaling is considered: i.e. $H = B = \sigma^2 = 1$, this upper bound reduces to:

$$\epsilon_{\text{L-SGD}} \leq O\left(\frac{1}{K^{\frac{1}{3}}R^{\frac{2}{3}}} + \frac{1}{\sqrt{MKR}}\right)$$

In this setting, the worst-case error of Minibatch SGD is:

$$\epsilon_{\text{MB-SGD}} = \Theta\left(\frac{1}{R} + \frac{1}{\sqrt{MKR}}\right).$$

One should notice that the denominator for Local SGD bound $K^{\frac{1}{3}}R^{\frac{2}{3}}$ dominates the denominator of Minibatch SGD guarantee R when K is sufficiently big compared to R.

Minibatch SGD can outperform Local SGD

In the general convex case authors also present the lower bound for the worst-case performance of Local SGD. When natural scaling is considered: i.e. $H = B = \sigma^2 = 1$, this lower bound reduces to:

$$\epsilon_{\text{L-SGD}} \geq O\left(\frac{1}{K^{\frac{2}{3}} R^{\frac{2}{3}}} + \frac{1}{\sqrt{MKR}}\right)$$

When comparing this with the worst-case error of Minibatch SGD , we find that local SGD performs worse than Minibatch SGD in the worst case when K is sufficiently small relative to R. This confirmed the concerns expressed in the previous section, in general convex setting in the worst case, Minibatch SGD is indeed superior to LocalSGD when number of local steps is small.

Implementation details

Local SGD and Minibatch SGD algorithms were implemented with Pytorch. Parallel execution was replaced by sequential (imitating parallel) due to big number of workers M which should be 50 or more for representative results.

For each M, K, and algorithm, the constant step size was tuned (via gridsearch on a logarithmic grid) to minimize the loss after r rounds of communication, individually for each $1 \leq r \leq R$.

Synthetic data

We generated a dataset of 50,000 points in $\mathbb{R}^{25}$, where the i-th coordinate of each point was independently sampled from a Gaussian distribution $\mathcal{N}(0, 10/i^2)$. The labels were generated based on the probability

$$P[y = 1 \mid x] = \sigma\left(\min\{\langle w_1^*, x\rangle + b_1^*, \langle w_2^*, x\rangle + b_2^*\}\right),$$

where $w_1^*, w_2^* \sim \mathcal{N}(0, I_{25\times 25})$ and $b_1^*, b_2^* \sim \mathcal{N}(0, 1)$. Each algorithm was used to train a linear model with a bias term by minimizing the binary cross-entropy loss over the dataset.

Charts for the synthetic data

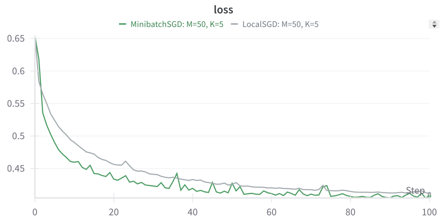

Figure: M = 50, K = 5: Minibatch SGD performs better.

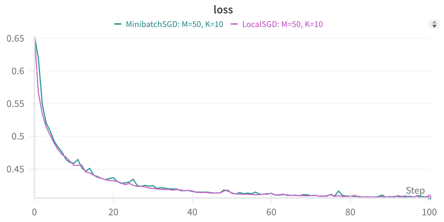

Figure: M = 50, K = 10: Identical performance.

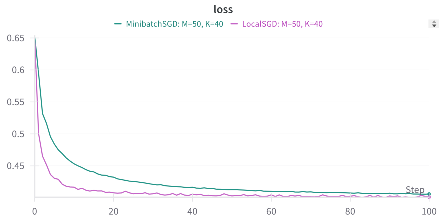

Figure: M = 50, K = 40: Local SGD is superior.

Smaller number of workers M

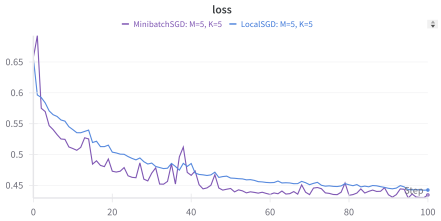

Figure: M = 5, K = 5: Minibatch SGD performs better.

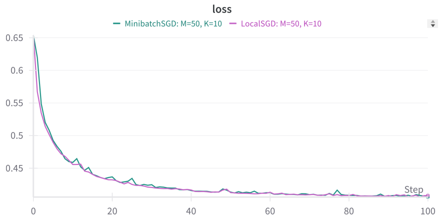

Figure: M = 5, K = 40: Local SGD is slightly better, but the difference is very small.

MNIST

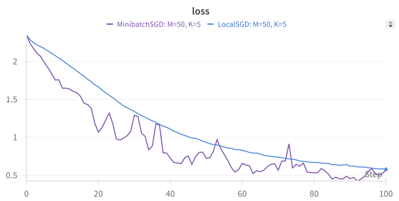

Figure: M=50, K=5: Minibatch does better but it is very noisy due to big learning rate.

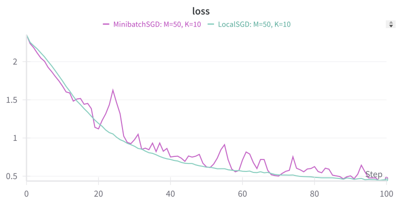

Figure: M=50, K=10: Local SGD is nearly the same as Minibatch but way less noisy.

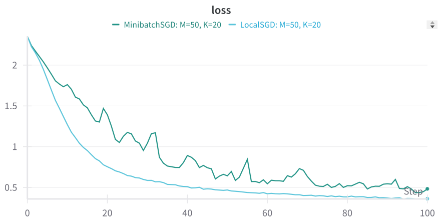

Figure: M=50, K=20: Local SGD performs significantly better than Minibatch SGD.

Thank you for your attention!

Natural Scaling

An important instance of (1) is a supervised learning problem where

$$f(x; z) = \ell(\langle x, \phi(z) \rangle, \text{label}(z))$$

is the loss on a single sample. When $|\ell'(x)|, |\ell''(x)| \leq 1$ (referring to derivatives with respect to the first argument), we have

$$H \leq |\ell''(x)| \|\phi(z)\|^2 \leq \|\phi(z)\|^2$$

and also

$$\sigma^2 \leq \|\nabla f\|^2 \leq |\ell'(x)|^2 \|\phi(z)\|^2 \leq \|\phi(z)\|^2$$

Thus, assuming that the upper bounds on $\ell'(x)$ and $\ell''(x)$ are comparable, the relative scaling of parameters which is considered as most "natural" is $H \approx \sigma^2$.