1. Problem Description

The problem is to minimize a sum of two convex functions,

$$\min_{x \in \mathbb{R}^d} \{ P(x) := f(x) + R(x) \},$$

where \( f \) is the average of a large number of smooth convex functions \( f_i(x) \), i.e.,

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$$

2. Assumptions

**Assumption 1.** The regularizer \( R : \mathbb{R}^d \to \mathbb{R} \cup \{ +\infty \} \) is convex and closed. The functions \( f_i : \mathbb{R}^d \to \mathbb{R} \) are differentiable and have Lipschitz continuous gradients with constant \( L > 0 \), i.e., \( \forall x, y \in \mathbb{R}^d, \| \nabla f_i(x) - \nabla f_i(y) \| \leq L \| x - y \| \), where \( \| \cdot \| \) is L2 norm.

**Assumption 2.** \( P \) is strongly convex with parameter \( \mu > 0 \), i.e., \( \forall x, y \in \text{dom}(P), \)

$$P(y) \geq P(x) + \langle \nabla P(x), y - x \rangle + \frac{\mu}{2} \| y - x \|^2,$$

where \( \partial P(x) \) is the subdifferential of \( P \) at \( x \).

3. The Algorithm (mS2GD)

**Algorithm 1** mS2GD

1. **Input:** \( m \) (max # of stochastic steps per epoch); \( h > 0 \) (stepsize); \( x_0 \in \mathbb{R}^d \) (starting point); minibatch size \( b \in \{1, \ldots, n\} \)
2. for \( k = 0, 1, 2, \ldots \) do
3. Compute and store \( y_k = -\nabla f(x_k) + \frac{\mu}{2} \sum_i \nabla f_i(x_k) \)
4. Initialize the inner loop: \( y_{k,0} \leftarrow x_k \)
5. Let \( t_k \leftarrow t \in \{1, 2, \ldots, m\} \) uniformly at random
6. for \( t = 0 \) to \( t_k - 1 \) do
7. Choose minibatch \( A_{k,t} \subset \{1, \ldots, n\} \) of size \( b \), uniformly at random
8. Compute a stochastic estimate of \( \nabla f_i(x_k) \): \( \nabla f_i(x_k) \leftarrow \frac{1}{b} \sum_{i \in A_{k,t}} \nabla f_i(x_k) - \nabla f_i(x_k) \)
9. \( y_{k,t+1} \leftarrow \text{prox}_{hP}(y_{k,t} - \mu y_{k,t}) \)
10. end for
11. Set \( x_{k+1} \leftarrow y_{k,t_k} \)
12. end for

This is a simplified case of our original algorithm. A complete version of the algorithm and convergence result, with known lower bounds of the convexity parameters \( \gamma_P, \gamma_R \) for \( F \) and \( R \) respectively, requires a non-uniform distribution for the number of steps per epoch [1].

4. Convergence Result

**Theorem 1.** Let Assumptions 1 and 2 be satisfied and let \( x^* \) be a minimizer of \( P(x) \). In addition, assume that the stepsize satisfies \( 0 < h < \min \{ \frac{1}{4L}, \frac{\mu}{L^2} \} \) and that \( m \) is sufficiently large so that

$$\rho = \frac{m_0}{m_0 - h} \leq 1,$$

where \( h = \min \{ \frac{1}{4L}, \frac{\mu}{L^2} \} \). Then mS2GD has linear convergence in expectation:

$$E(\| x_k - x^* \|) \leq \rho E(\| x_k - x^* \|) \text{ for } k > 0.$$

The following bound of variance is considered crucial:

$$\mathbb{E} \| x_k - x^* \|^2 \leq 4 \rho E(\| P(y_k) - P(x^*) \|).$$

5. Mini-Batch Speedup

**Theorem 2.** Fix target \( \rho \in (0, 1) \) and the mini-batch size \( b \). Let us define

$$h^b := \sqrt{\frac{1 + \rho}{\mu \rho} + 4 \rho \mu}.$$

Then the optimal step size \( h^b \) and the maximum size of inner loop \( m_t \) — which minimizes the number of gradient evaluations while keeping sufficient overall decrease — are given as follows:

If \( h^b < \frac{1}{2} \) then \( h_t^b = h^b \) and

$$m_t^b = \frac{8 \rho}{\mu \rho} \left( 1 + \sqrt{\frac{1}{4 \mu \rho^2} + (1 + \rho)^2} \right).$$

Otherwise \( h_t^b = \frac{1}{2} \) and \( m_t^b = \frac{1 - \rho \sqrt{1 + \rho^2}}{\mu \rho^2} \).

If \( m_t^b \leq m_t^b \), then we can reach the same accuracy with fewer gradient evaluations. Equation (4) shows that as long as the condition \( h^b < \frac{1}{2} \) is satisfied, \( m_t^b \) is decreasing at a rate roughly faster than \( \frac{1}{b} \). Hence, we can attain the same accuracy with less work, compared to the case when \( b = 1 \).

6. Numerical Experiments

**Figure 1:** rcv1 dataset, logistic regression, \( R(x) = \frac{1}{2} \| x \|^2 \).

**TOP:** Comparison between mS2GD and the other relevant algorithms implies its competitiveness. SGD+ is by using constant step-size in hindsight and SGD+ is the one with adaptive step-size \( h = h_{b}(k+1) \), where \( k \) is the number of effective passes.

**MIDDLE:** We compare mS2GD algorithm for different mini-batch sizes with the best parameters \( m \) and \( h \) for each batch size.

**BOTTOM:** We present the ideal speedup by parallelism — that would happen if we could always efficiently evaluate the \( b \) gradients in parallel, thus being \( b \) times faster.

7. References

