Semi-Stochastic Coordinate Descent

Jakub Konečný Zheng Qu Peter Richtárik

University of Edinburgh

NAIS Centre for Numerical Algorithms and Intelligent Software

1. Problem

Many problems in data science (e.g. machine learning, optimization and statistics) can be cast as loss minimization problems of the form

$$\min_{x \in \mathbb{R}^d} f(x), \qquad \text{where} \qquad f(x) = \frac{1}{n}\sum_{i=1}^{n} f_i(x). \tag{P}$$

We assume that each individual function $f_i : \mathbb{R}^d \to \mathbb{R}$ is convex and has Lipschitz continuous partial gradients with constants $\{L_{ij}\}_j$. That is,

$$\|\nabla_j f_i(x) - \nabla_j f_i(y)\| \le L_{ij}\|x - y\|, \ \ \forall x, y \in \mathbb{R}^d.$$

Further we assume that $f : \mathbb{R}^d \to \mathbb{R}$ is μ-strongly convex:

$$f(y) \ge f(x) + \langle \nabla f(x), y - x\rangle + \frac{\mu}{2}\|y - x\|^2, \ \ \forall x, y \in \mathbb{R}^d.$$

2. Related Methods

Gradient Descent

Update:	$x_{k+1} = x_k - h\nabla f(x_k)$
# of iterations:	$\mathcal{O}(\kappa \log(1/\epsilon)$
Cost of 1 iteration:	$\mathcal{O}(n)$

Stochastic gradient descent (SGD)

Update:	1. Sample $i \in \{1, \dots, n\} := [n]$ 2. $x_{k+1} = x_k - h_k \nabla f_i(x_k)$
# of iterations:	$\mathcal{O}(1/\epsilon)$
Cost of 1 iteration:	$\mathcal{O}(1)$

Coordinate descent (CD)

Update:	1. Sample $j \in \{1, \dots, d\} := [d]$ 2. $x_{k+1} = x_k - h_j \nabla_j f(x_k)$
# of iterations:	$\mathcal{O}(\kappa \log(1/\epsilon))$
Cost of 1 iteration:	$\mathcal{O}(\omega)$ ω — degree of partial separability

Semi-stochastic gradient descent (S2GD) [3]

Update: Outer loop: Compute and store $\nabla f(x_k)$ $y_{k,0} = x_k$

Inner loop:
1. Sample $i \in [n]$
2. $y_{k,t+1} = y_{k,t} - h\left(\nabla f_i(y_{k,t}) - \nabla f_i(x_k) + \nabla f(x_k)\right)$

$x_{k+1} = y_{k,t_k}$

# of iterations:	$\lceil \log(1/\epsilon) \rceil$
Cost of 1 iteration:	$\mathcal{O}(n + \kappa)$

3. Goal

SGD type of methods are often seen as sampling *rows* of a data matrix. Conversely, CD methods usually sample *columns* of the data matrix.
The aim of this work is to develop a hybrid of S2GD and CD, which efficiently samples both rows and columns of data. The method computes a stochastic estimate of the partial gradient $\nabla_j f_i(x)$ with variance diminishing property and updates only one coordinate at each iteration.

4. The S2CD algorithm [1]

S2CD Algorithm — Semi-Stochastic Coordinate Descent

parameters: m (max # of stochastic steps per epoch); $h > 0$ (stepsize parameter); $x_0 \in \mathbb{R}^d$ (starting point); set $\beta = \sum_{t=1}^{m}(1 - \mu h)^{m-t}$;
for $k = 0, 1, 2, \dots$ **do**
  Compute and store $\nabla f(x_k) = \frac{1}{n}\sum_i \nabla f_i(x_k)$;
  Initialize the inner loop: $y_{k,0} \leftarrow x_k$;
  Let $t_k = t \in [m]$ with probability $(1 - \mu h)^{m-t}/\beta$
  for $t = 0$ to $t_k - 1$ **do**
    Pick coordinate $j \in [d]$ with probability p_j
    Pick function index i from the set $\{i : L_{ij} > 0\}$ with probability q_{ij}
    $y_{k,t+1} \leftarrow y_{k,t} - hp_j^{-1}\left(\nabla_j f(x_k) + \frac{1}{nq_{ij}}\left(\nabla_j f_i(y_{k,t}) - \nabla_j f_i(x_k)\right)\right)e_j$;
  end for
  Reset the starting point: $x_{k+1} \leftarrow y_{k,t_k}$;
end for

The selection probability $\{p_j\}$ and $\{q_{ij}\}$ in S2CD are defined by:

$$p_j := \frac{\sum_{i=1}^{n}\omega_i L_{ij}}{\hat{L}}, \qquad q_{ij} := \frac{\omega_i L_{ij}}{\sum_{i=1}^{n}\omega_i L_{ij}},$$

where

$$\omega_i := |\{j : L_{ij} \neq 0\}|, \qquad \hat{L} := \frac{1}{n}\sum_{j=1}^{d}\sum_{i=1}^{n}\omega_i L_{ij}.$$

Special cases:

- When $n = 1$: S2CD reduces to a stochastic CD algorithm with importance sampling for the selection of $j \in [d]$.
- It is possible to extend S2CD to the case when coordinates are replaced by (nonoverlapping) blocks of coordinates. In such a setting, when all the variables form a single block, S2CD reduces to S2GD but with importance sampling for the selection of $i \in [n]$, as in [5].

Theorem 1

If $0 < h < 1/(2\hat{L})$ and m is sufficiently large so that

$$c := \frac{(1 - \mu h)^m}{(1 - (1 - \mu h)^m)(1 - 2\hat{L}h)} + \frac{2\hat{L}h}{1 - 2\hat{L}h} < 1,$$

then for all $k \ge 0$ we have:

$$\mathbf{E}[f(x_k) - f(x_*)] \le c^k\, \mathbf{E}[f(x_k) - f(x_*)].$$

Corollary: *Let $\hat{\kappa} := \hat{L}/\mu$. If we run the algorithm S2CD with stepsize h and m set as*

$$h = \frac{1}{(4e + 2)\hat{L}}, \qquad m \ge (4e + 2)\log(2e + 2)\,\hat{\kappa},$$

then for all $k \ge \lceil \log(1/\epsilon) \rceil$,

$$\mathbf{E}[f(x_k) - f(x_*)] \le \epsilon(f(x_0) - f(x_*)).$$

5. Complexity & comparison

Definition

We let $\mathcal{C}_{grad}$ be the average cost of evaluating the stochastic gradient ∇f_i and $\mathcal{C}_{pd}$ be the average cost of evaluating the stochastic partial derivative $\nabla_j f_i$.

S2CD complexity

The total work of S2CD can be written as

$$\mathcal{O}\left((n\mathcal{C}_{grad} + \hat{\kappa}\mathcal{C}_{pd})\log(1/\epsilon)\right).$$

The complexity results of methods such as S2GD/SVRG [3, 2, 5] and SAG/SAGA [4, 6] — in a similar but not identical setup to ours (some of these papers assume f_i to be L_i-smooth) — can be written in a similar form:

$$\mathcal{O}\left((n\mathcal{C}_{grad} + \kappa\mathcal{C}_{grad})\log(1/\epsilon)\right),$$

where $\kappa = L/\mu$ and either $L = L_{max}$ ([4, 2, 3, 6]), or $L = L_{avg} := \frac{1}{n}\sum_{i,j} L_{ij}$ ([5]).

The difference between our result and existing results is in the term $\hat{\kappa}\mathcal{C}_{pd}$ – previous results have $\kappa\mathcal{C}_{grad}$ in that place. This difference constitutes a trade-off: while $\hat{\kappa} \ge \kappa$, we clearly have $\mathcal{C}_{pd} \le \mathcal{C}_{grad}$. The comparison of the quantities $\kappa\mathcal{C}_{grad}$ and $\hat{\kappa}\mathcal{C}_{pd}$ is not straightforward and is problem dependent.

Conclusion

S2CD can be both better or worse than S2GD/SVRG/SAG/SAGA, depending on whether the increase of the condition number from κ to $\hat{\kappa}$ can or can not be compensated by the decrease of the derivative evaluation from $\mathcal{C}_{grad}$ to $\mathcal{C}_{pd}$.

6. References

[1] Konečný J., Qu Z., Richtárik P.: Semi-Stochastic Coordinate Descent, OPT 2014 @ NIPS

[2] Johnson R., Zhang T.: Accelerating Stochastic Gradient Descent using Predictive Variance Reduction, Advances in Neural Information Processing Systems, 2013

[3] Konečný J., Richtárik P.: Semi-Stochastic Gradient Descent Methods, 2013

[4] Schmidt M., Le Roux N., Bach F.: Minimizing Finite Sums with the Stochastic Average Gradient, 2013

[5] Xiao L., Zhang T.: A proximal stochastic gradient method with progressive variance reduction, 2014

[6] Defazio A., Bach F., Lacoste-Julien S.: SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives, 2014