Block stochastic gradient update method

Yangyang Xu[*] and Wotao Yin[†]

[*]IMA, University of Minnesota

[†] Department of Mathematics, UCLA

November 1, 2015

This work was done while in Rice University

Stochastic gradient method

Consider the stochastic programming

$$\min_{\mathbf{x}\in X} F(x) = \mathbb{E}_\xi f(\mathbf{x};\xi).$$

Stochastic gradient update (SG):

$$\mathbf{x}^{k+1} = \mathcal{P}_X\big(\mathbf{x}^k - \alpha_k \tilde{\mathbf{g}}^k\big)$$

- $\tilde{\mathbf{g}}^k$ a stochastic gradient, often $\mathbb{E}[\tilde{\mathbf{g}}^k] \in \partial F(\mathbf{x}^k)$
- Originally for stochastic problem where exact gradient not available
- Now also popular for deterministic problem where exact gradient expensive; e.g., $F(x) = \frac{1}{N}\sum_{i=1}^N f_i(\mathbf{x})$ with large N
- Faster than deterministic gradient method to reach not-high accuracy

Stochastic gradient method

- First appears in [Robbins-Monro'51]; now tons of works
- $\mathcal{O}(1/\sqrt{k})$ rate for weakly convex problem and $\mathcal{O}(1/k)$ for strongly convex problem (e.g., [Nemirovski et. al'09])
- For deterministic problem, linear convergence is possible if exact gradient allowed periodically [Xiao-Zhang'14]
- Convergence in terms of first-order optimality condition for nonconvex problem [Ghadimi-Lan'13]

Block gradient descent

Consider the problem

$$\min_{\mathbf{x}} F(\mathbf{x}) = f(\mathbf{x}_1, \ldots, \mathbf{x}_s) + \sum_{i=1}^{s} r_i(\mathbf{x}_i).$$

- f is smooth
- r_i's possibly nonsmooth and extended-valued

Block gradient update (BGD):

$$\mathbf{x}_{i_k}^{k+1} = \arg\min_{\mathbf{x}_{i_k}} \langle \nabla_{i_k} f(\mathbf{x}^k), \mathbf{x}_{i_k} - \mathbf{x}_{i_k}^k \rangle + \frac{1}{2\alpha_k} \|\mathbf{x}_{i_k} - \mathbf{x}_{i_k}^k\|_2^2 + r_{i_k}(\mathbf{x}_{i_k})$$

- Simpler than classic block coordinate descent, i.e., block minimization
- Allows different ways to choose i_k: cyclicly, greedily, or randomly
- Low iteration complexity
- Larger stepsize than full gradient and often faster convergence

Block gradient descent

- First appears in [Tseng-Yun'09]; famous since [Nesverov'12]
- Both cyclic and randomized selection: $\mathcal{O}(1/k)$ for weakly convex problem and linear convergence for strongly convex problem (e.g., [Hong et. al'15])
- Cyclic version harder than random or greedy version to analyze
- Subsequence convergence for nonconvex problem and whole sequence convergence if certain local property holds (e.g., [X.-Yin'14])

Stochastic programming with block structure

Consider problem

$$\min_{\mathbf{x}} \Phi(\mathbf{x}) = \mathbb{E}_\xi f(\mathbf{x}_1, \ldots, \mathbf{x}_s; \xi) + \sum_{i=1}^{s} r_i(\mathbf{x}_i) \qquad \text{(BSP)}$$

- Example: tensor regression [Zhou-Li-Zhu'13]

$$\min_{X_1,\ldots,X_s} \mathbb{E}[\ell(X_1 \circ \cdots \circ X_s; \boldsymbol{\mathcal{A}}, b)]$$

This talk presents an algorithm for (BSP) **with properties:**

- Only requiring stochastic block gradient
- Simple update and low computational complexity
- Guaranteed convergence
- Optimal convergence rate if the problem is convex

How and why

- use stochastic partial gradient in BGD
  - exact partial gradient unavailable or expensive
  - stochastic gradient works but performs not as well
- random cyclic selection, i.e., shuffle and then cycle
  - random shuffling for faster convergence and more stable performance [Chang-Hsieh-Lin'08]
  - cyclic for lower computational complexity but analysis more difficult

Block stochastic gradient method

At each iteration/cycle k

1. Sample one function or a batch of functions
2. Random shuffle blocks to $(k_1, \ldots, k_s)$
3. From $i = 1$ through s, do

$$\mathbf{x}_{k_i}^{k+1} = \arg\min_{\mathbf{x}_{k_i}} \langle \tilde{\mathbf{g}}_{k_i}^k, \mathbf{x}_{k_i} \rangle + \frac{1}{2\alpha_{k_i}^k} \|\mathbf{x}_{k_i} - \mathbf{x}_{k_i}^k\|^2 + r_{k_i}(\mathbf{x}_{k_i})$$

- $\tilde{\mathbf{g}}_{k_i}^k$ stochastic partial gradient, dependent on sampled functions and intermediate point $(\mathbf{x}_{k_{<i}}^{k+1}, \mathbf{x}_{k_{\geq i}}^k)$
- possibly biased estimate, i.e., $\mathbb{E}[\tilde{\mathbf{g}}_{k_i}^k - \nabla F(\mathbf{x}_{k_{<i}}^{k+1}, \mathbf{x}_{k_{\geq i}}^k)] \neq 0$, where $F(\mathbf{x}) = \mathbb{E}_\xi f(\mathbf{x}; \xi)$

Pros and cons of cyclic selection

Pros:

- lower computational complexity, e.g., for $\Phi(\mathbf{x}) = \mathbb{E}_{(\mathbf{a},b)}(\mathbf{a}^\top \mathbf{x} - b)^2$ with $\mathbf{x} \in \mathbb{R}^n$
  - cyclic selection takes about $2n$ to update all coordinates once
  - random selection takes n to update one coordinate
- Gauss-seidel type fast convergence (see numerical results later)

Cons:

- biased stochastic partial gradient
  - makes analysis more difficult

Literature

Just a few papers so far

- [Liu-Wright, arXiv14]: an asynchronous parallel randomized Kaczmarz algorithm
- [Dang-Lan, SIOPT15]: stochastic block mirror descent methods for nonsmooth and stochastic optimization
- [Zhao et al. NIPS14]: accelerated mini-batch randomized block coordinate descent method
- [Wang-Banerjee, arXiv14]: randomized block coordinate descent for online and stochastic optimization
- [Hua-Kadomoto-Yamashita, OptOnline15]: regret analysis of block coordinate gradient methods for online convex programming

Assumptions

Recall $F(\mathbf{x}) = \mathbb{E}_\xi f(\mathbf{x}; \xi)$. Let $\boldsymbol{\delta}_i^k = \tilde{\mathbf{g}}_i^k - \nabla_{\mathbf{x}_i} F(\mathbf{x}_{<i}^{k+1}, \mathbf{x}_{\geq i}^k)$.

Error bound of stochastic partial gradient:

$$\left\|\mathbb{E}[\boldsymbol{\delta}_i^k | \mathbf{x}^{k-1}]\right\| \leq A \cdot \max_j \alpha_j^k,\ \forall i, k,\ (A = 0 \text{ if unbiased})$$

$$\mathbb{E}\|\boldsymbol{\delta}_i^k\|^2 \leq \sigma_k^2 \leq \sigma^2, \forall i, k.$$

Lipschitz continuous partial gradient:

$$\|\nabla_{\mathbf{x}_i} F(\mathbf{x} + (0, \ldots, \mathbf{d}_i, \ldots, 0)) - \nabla_{\mathbf{x}_i} F(\mathbf{x})\| \leq L_i \|\mathbf{d}_i\|, \forall i, \forall \mathbf{x}, \mathbf{d}.$$

Convergence of block stochastic gradient

- **Convex case:** F and r_i's are convex. Take $\alpha_i^k = \alpha_k = \mathcal{O}(\frac{1}{\sqrt{k}}), \forall i, k$, and let

$$\tilde{\mathbf{x}}^k = \frac{\sum_{\kappa=1}^{k} \alpha_\kappa \mathbf{x}^{\kappa+1}}{\sum_{\kappa=1}^{k} \alpha_\kappa}.$$

  Then

$$\mathbb{E}[\Phi(\tilde{\mathbf{x}}^k) - \Phi(\mathbf{x}^*)] \leq \mathcal{O}(\log k/\sqrt{k}).$$

  - Can be improved to $\mathcal{O}(1/\sqrt{k})$ if the number of iterations is pre-known, and thus achieves the optimal order of rate

- **Strongly convex case:** Φ is strongly convex. Take $\alpha_i^k = \alpha_k = \mathcal{O}(\frac{1}{k}), \forall i, k$. Then

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^*\|^2 \leq \mathcal{O}(1/k).$$

  - Again, optimal order of rate is achieved

Convergence of block stochastic gradient

- **Unconstrained smooth nonconvex case**: If $\{\alpha_i^k\}$ is taken such that

$$\sum_{k=1}^{\infty} \alpha_i^k = \infty,\ \sum_{k=1}^{\infty} (\alpha_i^k)^2 < \infty,\ \forall i,$$

  then

$$\lim_{k\to\infty} \mathbb{E}\|\nabla\Phi(\mathbf{x}^k)\| = 0.$$

- **Nonsmooth nonconvex case**: If $\alpha_i^k < \frac{2}{L_i}, \forall i, k$, and $\sum_{k=1}^{\infty} \sigma_k^2 < \infty$, then

$$\lim_{k\to\infty} \mathbb{E}\Big[\mathsf{dist}(\mathbf{0}, \partial\Phi(\mathbf{x}^k))\Big] = 0.$$

Numerical experiments

Tested problems

- Stochastic least square
- Linear logistic regression
- Bilinear logistic regression
- Low-rank tensor recovery from Gaussian measurements

Tested methods

- block stochastic gradient (BSG) [proposed]
- block gradient (deterministic)
- stochastic gradient method (SG)
- stochastic block mirror descent (SBMD) [Dang-Lan'15]

Stochastic least square

Consider $\quad \min_{\mathbf{x}} \mathbb{E}_{(\mathbf{a},b)} \frac{1}{2}(\mathbf{a}^\top \mathbf{x} - b)^2$

- $\mathbf{a} \sim \mathcal{N}(0, \mathbf{I})$, $b = \mathbf{a}^\top \hat{\mathbf{x}} + \eta$, and $\eta \sim \mathcal{N}(0, 0.01)$
- $\hat{\mathbf{x}}$ is the optimal solution, and minimum value 0.005
- $\{(\mathbf{a}_i, b_i)\}$ observed sequentially from $i = 1$ to N
- Deterministic (partial) gradient unavailable

Objective values by different methods

N Samples	BSG	SG	SBMD-10	SBMD-50	SBMD-100
4000	6.45e-3	**6.03e-3**	67.49	4.79	1.03e-1
6000	**5.69e-3**	5.79e-3	53.84	1.43	1.43e-2
8000	**5.57e-3**	5.65e-3	42.98	4.92e-1	6.70e-3
10000	**5.53e-3**	5.58e-3	35.71	2.09e-1	5.74e-3

- One coordinate as one block
- SBMD-t: SBMD with t coordinates selected each update
- Objective valued by another 100,000 samples
- Each update of all methods costs $\mathcal{O}(n)$

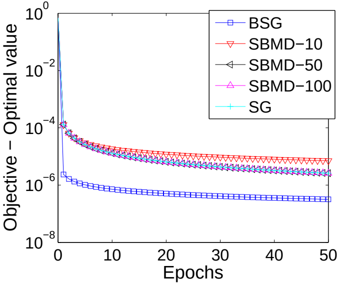
Observation: Better to update more coordinates and BSG performs best

Logistic regression

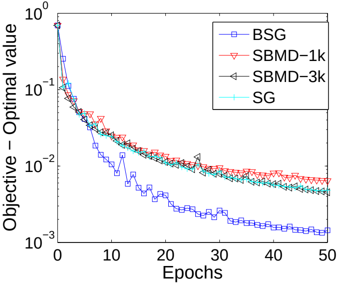
$$\boxed{\min_{\mathbf{w},b} \frac{1}{N} \sum_{i=1}^{N} \log\Big(1 + \exp\big[-y_i\big(\mathbf{x}_i^\top \mathbf{w} + b\big)\big]\Big) \qquad \text{(LR)}}$$

- Training samples $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$ with $y_i \in \{-1, +1\}$
- Deterministic problem but exact gradient expensive for large N
- Stochastic gradient faster than deterministic gradient for not-high accuracy

Performance of different methods on logistic regression



(a) random dataset



(b) gisette dataset

- Random dataset: 2000 Gaussian random samples of dimension 200
- gisette dataset: 6,000 samples of dimension 5000, from LIBSVM Datasets

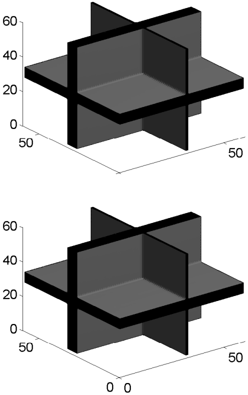
Observation: BSG gives best performance among compared methods.

Low-rank tensor recovery from Gaussian measurements

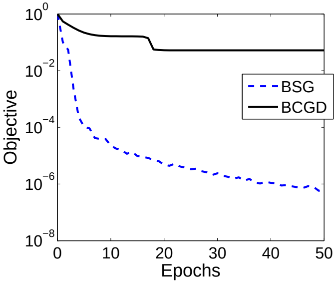
$$\boxed{\min_{\mathbf{X}} \frac{1}{2N} \sum_{\ell=1}^{N} (\mathcal{A}_\ell(\mathbf{X}_1 \circ \mathbf{X}_2 \circ \mathbf{X}_3) - b_\ell)^2 \qquad \text{(LRTR)}}$$

- $b_\ell = \mathcal{A}_\ell(\boldsymbol{\mathcal{M}}) = \langle \boldsymbol{\mathcal{G}}_\ell, \boldsymbol{\mathcal{M}} \rangle$ with $\boldsymbol{\mathcal{G}}_\ell \sim \mathcal{N}(0, \boldsymbol{\mathcal{I}})$, $\forall \ell$
- $\boldsymbol{\mathcal{G}}_\ell$'s are dense
- For large N, reading all $\boldsymbol{\mathcal{G}}_\ell$ may out of memory even for medium $\boldsymbol{\mathcal{G}}_\ell$
- Deterministic problem but exact gradient too expensive for large N

Peformance of block deterministic and stochastic gradient



- $\mathcal{G}_\ell \in \mathbb{R}^{60\times 60\times 60}$ and $N = 40,000$
- Original (top) and recovered (bottom) by BSG with 50 epochs



- $\mathcal{G}_\ell \in \mathbb{R}^{32\times 32\times 32}$ and $N = 15,000$
- BCGD: block deterministic gradient
- BSG: block stochastic gradient

Observation: BSG faster and BCGD trapped at bad local solution

Bilinear logistic regression

$$\min_{\mathbf{U},\mathbf{V},b} \frac{1}{N} \sum_{i=1}^{N} \log\Big(1 + \exp\big[-y_i\big(\langle \mathbf{U}\mathbf{V}^\top, \mathbf{X}_i\rangle + b\big)\big]\Big) \qquad \text{(BLR)}$$

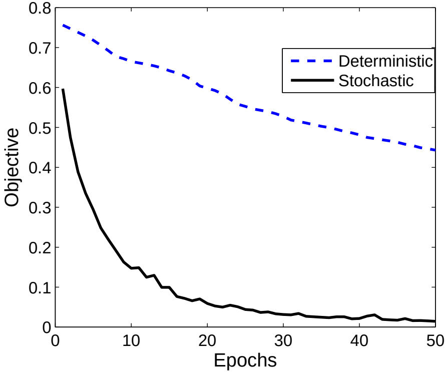
- Training samples $\{(\mathbf{X}_i, y_i)\}_{i=1}^N$ with $y_i \in \{-1, +1\}$
- Better than linear logistic regression for 2D dataset [Dyrholm et al.'07]

BCI competition EEG dataset

- Recorded from a healthy person using 118 channels;
- Visual cues (letter presentation) were shown;
- Performed: left hand, right foot, or tongue;
- 2100 marked data points of "left hand" and "right foot" were used;
- Each data point is 118×100

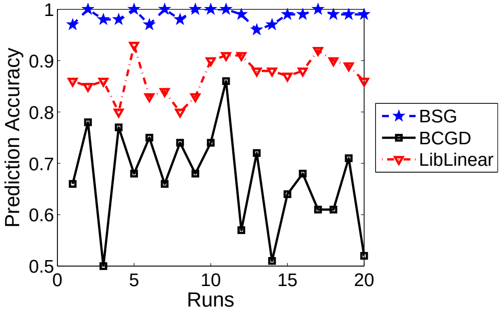
http://www.bbci.de/competition/iii/

Performance of block deterministic and stochastic gradient



Observation: stochastic method faster than deterministic one

Performance of linear and bilinear logistic regression



- Each run, 2000 for training and 100 for testing
- BSG and BCGD run to 30 epochs
- LibLinear solves linear logistic regression

Observation: bilinear model better than linear one on EEG data

Conclusions

- Proposed a block stochastic gradient method for stochastic programming
  - Combines block gradient and stochastic gradient methods
  - Inherits both advantages and better than either one individually
- Analyzed its convergence and rate
  - Optimal order of convergence rate for convex problems
  - Convergence in terms of first-order optimality condition for nonconvex problems
- Tested on both convex and nonconvex problems
  - stochastic least square
  - linear and bilinear logistic regression
  - low-rank tensor recovery from dense Gaussian measurements

References

- Y. Xu and W. Yin. Block stochastic gradient iteration for convex and nonconvex optimization. SIOPT15.
- J. Shi, Y. Xu and R. Baraniuk. Sparse bilinear logistic regression. arXiv14.
- C. Dang and G. Lan. Stochastic block mirror descent methods for nonsmooth and stochastic optimization. SIOPT15.
- Y. Xu and W. Yin. A block coordinate descent method for regularized multi-convex optimization with applications to nonnegative tensor factorization and completion. SIIMS13.
- A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro. Robust stochastic approximation approach to stochastic programming, SIOPT09.