Optimization Theory and Algorithm	**Lecture 13 - 10/29/2021**

Lecture 13

Lecturer: Xiangyu Chang *Scribe: Xiangyu Chang*

Edited by: Junbo Hao

1 BCD

Example 1.1. Let us consider the problem

$$\min f(x,y) = x^2 - 2xy + 10y^2 - 4x - 20y.$$

If we fix y, then $\nabla_x f(x,y) = 2x - 4y - 4 = 0$, that is x=y+2. If we fix x, then $\nabla_y f(x,y) = 20y - 2x - 20 = 0$, that is $y = x/10 + 1$.

$$\begin{cases} x^{t+1} = y^t + 2, \\ y^{t+1} = x^t/10 + 1. \end{cases}$$

Algorithm 1 Block Coordinate Descent

1: **Input:** Given a initial starting point $\mathbf{x}^0 = (\mathbf{x}_1^0, \ldots, \mathbf{x}_K^0) \in \mathbb{R}^n$, and $t = 0$
2: **for** $t = 0, 1, \ldots, T$ **do**
3:   **for** $k = 0, 1, \ldots, K$ **do**
4:     Do (i) or (ii) or (iii) for Eq.(1).
5:   **end for**
6: **end for**
7: **Output:** $\mathbf{x}^T$.

$$\min_{\mathbf{x}} f(\mathbf{x}) = f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_K) + \sum_{k=1}^{K} r_k(\mathbf{x}_k), \tag{1}$$

Remark 1.2.
- *This algorithm is called "Block Coordinate Descent". If $K = n$, it also called "Coordinate Descent".*
- *This algorithm does not always convert to the optimal solution.*
- *The related convergence theory can be found in two review papers [Wri15, STXY16].*

Example 1.3. (Group LASSO)

Suppose that $\mathbf{x} = (x_1, \ldots, x_n)^\top \in \mathbb{R}^n = (\mathbf{z}_1, \ldots, \mathbf{z}_K)^\top$ and $\mathbf{z}_k \in \mathbb{R}^{n_k}, \sum_{k=1}^K n_k = n, A = [A_1, A_2, \ldots, A_K] \in \mathbb{R}^{m\times n}$. Then Group LASSO is

$$\min_{\mathbf{x}} \frac{1}{2}\|A\mathbf{x} - \mathbf{b}\|^2 + \lambda \sum_{k=1}^K \|\mathbf{z}_k\|_2,$$

where $\|\mathbf{z}_k\|_2 = \sqrt{\sum_{l=1}^{n_k} z_{kl}^2}$ This is equivalent to

$$\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{b} - \sum_{k=1}^K A_k\mathbf{z}_k\|^2 + \lambda \sum_{k=1}^K \|\mathbf{z}_k\|_2. \tag{2}$$

BCD algorithm: Given $\mathbf{z}_2^t, \ldots, \mathbf{z}_K^t$, then let $\mathbf{b}^t = \mathbf{b} - \sum_{k=2}^K A_k\mathbf{z}_k^t$. Then Eq.(2) is equivalent to

$$\min_{\mathbf{z}_1} \frac{1}{2}\|\mathbf{b}^t - A_1\mathbf{z}_1\|^2 + \lambda\|\mathbf{z}_1\|_2.$$

If $\mathbf{z}_1 \neq 0$, then $-A_1^\top(\mathbf{b}^t - A_1\mathbf{z}_1) + \lambda\frac{\mathbf{z}_1}{\|\mathbf{z}_1\|_2} = 0$, so,

$$\mathbf{z}_1 = (A_1^\top A_1 + \frac{\lambda I}{\|\mathbf{z}_1\|_2})^{-1}A_1^\top\mathbf{b}^t.$$

The iterative step is

$$\mathbf{z}_1^{t+1} \leftarrow (A_1^\top A_1 + \frac{\lambda I}{\|\mathbf{z}_1^t\|_2})^{-1}A_1^\top\mathbf{b}^t.$$

If $\mathbf{z}_1 = 0$, then $0 \in \partial(\frac{1}{2}\|\mathbf{b}^t - A_1\mathbf{z}_1\|^2 + \lambda\|\mathbf{z}_1\|_2) = -A_1^\top\mathbf{b}^t + \lambda s$, where $s \in \partial\|0\|_2 = \{s \mid \|s\|_2 \leqslant 1\}$.

Thus, $\|A_1^\top\mathbf{b}^t\| \leqslant \lambda$. Final update is

$$\mathbf{z}_1^{t+1} \leftarrow \begin{cases} 0, & \text{if } \|A_1^\top\mathbf{b}^t\| \leqslant \lambda, \\ (A_1^\top A_1 + \frac{\lambda I}{\|\mathbf{z}_1^t\|_2})^{-1}A_1^\top\mathbf{b}^t, & \text{otherwise.} \end{cases}$$

Example 1.4. (K-means)

Suppose we have a data matrix $A_{m\times n} = (\mathbf{a}_1^\top, \ldots, \mathbf{a}_m^\top)^\top$. We introduce a corresponding binary indicator variable $r_{ik} \in \{0, 1\}, i \in [m], k \in [K]$ to describe which of the k clusters the data point $\mathbf{a}_i$ is assigned. If $\mathbf{a}_i$ is assigned to cluster k, then $r_{ik} = 1$, otherwise $r_{ik'} = 0, k' \neq k$. Let μ_k be the mean vector of cluster k, then the objective function of K-means is

$$\min_{\mu_k, r_{ik}} \sum_{i=1}^m \sum_{k=1}^K r_{ik}\|\mathbf{a}_i - \mu_k\|^2 = \ell(R, \mu), \tag{3}$$

where R includes all the indicator variables and μ includes all μ_k.

K-means Algorithm:

- Fix r_{ik}, $\nabla_{\mu_k}\ell(R, \mu) = -2\sum_{i=1}^m r_{ik}(\mathbf{a}_i - \mu_k) = 0$, that is

$$\mu_k = \frac{\sum_{i=1}^m r_{ik}\mathbf{a}_i}{\sum_{i=1}^m r_{ik}}.$$

- Fix μ_k then,

$$r_{ik^*} = \begin{cases} 1, & \text{if } k^* = \arg\min_{1 \leqslant k \leqslant K} \|\mathbf{a}_i - \mu_k\|^2, \\ 0, & \text{otherwise.} \end{cases}$$

We further denote $H = (\mu_1^\top, \mu_2^\top, \ldots, \mu_K^\top)^\top \in \mathbb{R}^{K \times n}$ and $R = (r_1^\top, \ldots, r_m^\top)^\top \in \mathbb{R}^{m \times K}$, then the objective function of K-means can be reformulated as:

$$\min_{R,H} \|A - RH\|_F^2.$$

The K-means algorithm first fixes R to solve H, then fixes H to solve R respectively.

2 SVRG

How to reduce the variance of stochastic gradient? Let us consider an important method in the MCMC method. We try to estimate the unknown expectation $\bar{\mathbf{x}}$ of a random variable $\mathbf{x}$ and that we have access to another random variable, $\mathbf{z}$, whose expectation $\bar{\mathbf{z}}$ is known. The the quantity $\mathbf{x}_\mathbf{z} = \mathbf{x} - \mathbf{z} + \bar{\mathbf{z}}$ has expectation $\bar{\mathbf{x}}$ and variance

$$V(\mathbf{x}_\mathbf{z}) = V(\mathbf{x}) + V(\mathbf{z}) - 2\mathrm{Cov}(\mathbf{x}, \mathbf{z}) \tag{4}$$

where $V(\cdot)$ is the variance and $\mathrm{Cov}(\cdot, \cdot)$ is the covariance. Then $V[\mathbf{x}_\mathbf{z}]$ is lower than $V[\mathbf{x}]$ whenever $\mathbf{z}$ is sufficiently positively correlated with $\mathbf{x}$ and the variance reduction is larger when the control variate is more correlated with the random variable.

So what $\mathbf{z}$ should we choose to reduce the variance of stochastic gradient estimation? That is

$$\widetilde{g}_i(\mathbf{x}_t) = g_i(\mathbf{x}_t) - z_i(\mathbf{x}_t) + \frac{1}{N}\sum_{j=1}^{N} \mathbf{z}_j(\mathbf{x}_t) \tag{5}$$

Let us first refer to Algorithm 2.

Now we bound the variance of stochastic gradient.

Lemma 2.1. *Denote that,*

$$\mathbf{v}_t = \nabla f_{i_t}(\mathbf{x}_{t-1}) - \nabla f_{i_t}(\widetilde{\mathbf{x}}) + \widetilde{\mathbf{z}} \tag{7}$$

It holds that

$$\mathbb{E}\|\mathbf{v}_t\|^2 \leqslant 4L[f(\mathbf{x}_{t-1}) - f(\mathbf{x}^*) + f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] \tag{8}$$

Proof. Given any i, consider

$$h_i(\mathbf{x}) = f_i(\mathbf{x}) - f_i(\mathbf{x}^*) - \nabla^\top f_i(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*), \text{Bregman divergence} \tag{9}$$

Algorithm 2 SVRG

Parameters update frequency T and learning rate η

Initialize $\widetilde{\mathbf{x}}_0$

for $s = 1, 2, \ldots$ **do**

$\widetilde{\mathbf{x}} = \widetilde{\mathbf{x}}_{s-1}$

$\widetilde{\mathbf{z}} = \frac{1}{m}\sum_{i=1}^{m} \nabla f_i(\widetilde{\mathbf{x}})$

$x_0 = \widetilde{\mathbf{x}}$

for $t = 1, 2, \ldots, T$ **do**

Randomly pick $i_t \in \{1, \ldots, m\}$ and update weight

$$\mathbf{x}_t = \mathbf{x}_{t-1} - \eta\left(\nabla f_{i_t}(\mathbf{x}_{t-1}) - \nabla f_{i_t}(\widetilde{\mathbf{x}}) + \widetilde{\mathbf{z}}\right) \tag{6}$$

end for

Set $\widetilde{\mathbf{x}}_s = \mathbf{x}_t$ for randomly chosen $t \in \{0, \ldots, T-1\}$

end for

We know that $h_i(\mathbf{x}^*) = \min_w h_i(\mathbf{w})$ since $\nabla h_i(\mathbf{x}^*) = 0$. Therefore

$$0 = h_i(\mathbf{x}^*) \leqslant \min_{\eta}\left[h_i(\mathbf{x} - \eta \nabla h_i(\mathbf{x}))\right] \tag{10}$$

$$\leqslant \min_{\eta}\left[h_i(\mathbf{x}) - \eta\|\nabla h_i(\mathbf{x})\|^2 + 0.5L\eta^2\|\nabla h_i(\mathbf{x})\|^2\right] \tag{11}$$

$$= h_i(\mathbf{x}) - \frac{1}{2L}\|\nabla h_i(\mathbf{x})\|^2. \tag{12}$$

That is,

$$\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{x}^*)\|^2 \leqslant 2L(f_i(\mathbf{x}) - f_i(\mathbf{x}^*) - \nabla^\top f_i(\mathbf{x}^*)(\mathbf{x} - \mathbf{x}^*)) \tag{13}$$

By summing the above inequality over $i = 1, \ldots, n$, and using the fact that $\nabla f(\mathbf{x}^*) = 0$, we obtain that

$$\frac{1}{n}\sum_{i=1}^{n}\|\nabla f_i(\mathbf{x}) - \nabla f_i(\mathbf{x}^*)\|^2 \leqslant 2L(f(\mathbf{x}) - f(\mathbf{x}^*)) \tag{14}$$

Let us denote

$$\mathbf{v}_t = \nabla f_{i_t}(\mathbf{x}_{t-1}) - \nabla f_{i_t}(\widetilde{\mathbf{x}}) + \widetilde{\mathbf{z}} \tag{15}$$

Conditioned on $\mathbf{x}_{t-1}$, we can take expectation with respect to i_t, and obtain that

$$\mathbb{E}\|\mathbf{v}_t\|^2 \leqslant 2\mathbb{E}\|\nabla f_{i_t}(\mathbf{x}) - \nabla f_{i_t}(\mathbf{x}^*)\|^2 + 2\mathbb{E}\|[\nabla f_{i_t}(\widetilde{\mathbf{x}}) - \nabla f_{i_t}(\mathbf{x}^*)] - \nabla f(\widetilde{\mathbf{x}})\|^2 \tag{16}$$

$$= 2\mathbb{E}\|\nabla f_{i_t}(\mathbf{x}) - \nabla f_{i_t}(\mathbf{x}^*)\|^2 + 2\mathbb{E}\|[\nabla f_{i_t}(\widetilde{\mathbf{x}}) - \nabla f_{i_t}(\mathbf{x}^*)] - \mathbb{E}[\nabla f_{i_t}(\widetilde{\mathbf{x}}) - \nabla f_{i_t}(\mathbf{x}^*)]\|^2 \tag{17}$$

$$\leqslant 2\mathbb{E}\|\nabla f_{i_t}(\mathbf{x}) - \nabla f_{i_t}(\mathbf{x}^*)\|^2 + 2\mathbb{E}\|[\nabla f_{i_t}(\widetilde{\mathbf{x}}) - \nabla f_{i_t}(\mathbf{x}^*)]\|^2 \tag{18}$$

$$\leqslant 4L[f(\mathbf{x}_{t-1}) - f(\mathbf{x}^*) + f(\widetilde{\mathbf{x}}) - f(x^*)] \tag{19}$$

■

Theorem 2.2. *The sequence* $\{\widetilde{\mathbf{x}}_s\}$ *in Algorithm 2 has the following property*

$$\mathbb{E}[f(\widetilde{\mathbf{x}}_s) - f(\mathbf{x}^*)] \leqslant \left[\frac{1}{\mu\eta(1-2L\eta)T} + \frac{2L\eta}{1-2L\eta}\right] \mathbb{E}[f(\widetilde{\mathbf{x}}_{s-1}) - f(\mathbf{x}^*)] \tag{20}$$

Proof. By conditioning on $\mathbf{x}_{t-1}$, we have $\mathbb{E}\mathbf{v}_t = \nabla f(\mathbf{x}_{t-1})$ and this leads to

$$\begin{aligned}
\mathbb{E}\|\mathbf{x}_t - \mathbf{x}^*\|^2 =& \|\mathbf{x}_{t-1} - \mathbf{x}^*\|^2 - 2\eta(\mathbf{x}_{t-1} - \mathbf{x}^*)^\top \mathbb{E}\mathbf{v}_t + \eta^2 \mathbb{E}\|\mathbf{v}_t\|^2 & (21)\\
\leqslant& \|\mathbf{x}_{t-1} - \mathbf{x}^*\|^2 - 2\eta(\mathbf{x}_{t-1} - \mathbf{x}^*)^\top \nabla f(\mathbf{x}_{t-1}) + 4L\eta^2[f(\mathbf{x}_{t-1}) - f(\mathbf{x}^*) + f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (22)\\
=& \|\mathbf{x}_{t-1} - \mathbf{x}^*\|^2 - 2\eta(1-2L\eta)[f(\mathbf{x}_{t-1} - f(\mathbf{x}^*)] + 4L\eta^2[f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (23)
\end{aligned}$$

We consider a fixed stage s, so that $\widetilde{\mathbf{x}} = \widetilde{\mathbf{x}}_{s-1}$ and $\widetilde{\mathbf{x}}_s$ is selected after all of the updates have completed. By summing the previous inequality over $t = 1, \ldots, T$, taking expectation with all the history, we obtain that

$$\begin{aligned}
& \mathbb{E}\|\mathbf{x}_T - \mathbf{x}^*\| + 2\eta(1-2L\eta)T\mathbb{E}[f(\widetilde{\mathbf{x}}_s - f(\mathbf{x}^*)] & (24)\\
\leqslant& \mathbb{E}\|\mathbf{x}_0 - \mathbf{x}^*\|^2 + 4LT\eta^2\mathbb{E}[f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (25)\\
=& \mathbb{E}\|\widetilde{\mathbf{x}} - \mathbf{x}^*\|^2 + 4LT\eta^2\mathbb{E}[f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (26)\\
\leqslant& \frac{2}{\mu}\mathbb{E}[f(\widetilde{\mathbf{x}}) - f(x^*)] + 4LT\eta^2\mathbb{E}[f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (27)\\
=& 2(\mu^{-1} + 2LT\eta^2)\mathbb{E}[f(\widetilde{\mathbf{x}}) - f(\mathbf{x}^*)] & (28)
\end{aligned}$$

We thus obtain that

$$\mathbb{E}[f(\widetilde{\mathbf{x}}_s) - f(\mathbf{x}^*)] \leqslant \left[\frac{1}{\mu\eta(1-2L\eta)T} + \frac{2L\eta}{1-2L\eta}\right] \mathbb{E}[f(\widetilde{\mathbf{x}}_{s-1}) - f(\mathbf{x}^*)] \tag{29}$$

■

References

[STXY16] Hao-Jun Michael Shi, Shenyinying Tu, Yangyang Xu, and Wotao Yin. A primer on coordinate descent algorithms. *arXiv preprint arXiv:1610.00040*, 2016.

[Wri15] Stephen J Wright. Coordinate descent algorithms. *Mathematical Programming*, 151(1):3–34, 2015.