9 September - 15 September

1 Error bounds for inexact Gradient Descent

1.1 Getting rid of error term

Let us consider the following situation that is possible to happen during the convergence analysis. Consider a sequence $h_k \geq 0$ for which we want to prove a linear convergence to 0

$$h_k \leq (1-\alpha)^k C_0 \tag{1}$$

for some $\alpha \in (0,1)$ and $C_0 > 0$.

But sometimes in practice we have a weaker condition

$$h_k = h_k(\varepsilon_k) \leq (1-\beta)h_{k-1} + \varepsilon_k(c_1 + c_2\sqrt{h_{k-1}}) \tag{2}$$

where c_1 and c_2 are some positive constants and sequence $\varepsilon_k > 0$. How to select sequence ε_k to guarantee (1) for some α if h_{k-1} is unknown?

Let us first consider, that h_{k-1} is known, then selecting

$$\varepsilon_k \leq \frac{(\beta-\alpha)h_{k-1}}{c_1 + c_2\sqrt{h_{k-1}}} \tag{3}$$

we have

$$h_k \leq (1-\alpha)h_{k-1} \leq (1-\alpha)^k h_0. \tag{4}$$

But usually we have no knowledge about it.

Let us prove (1) by mathematical induction.

Base: $h_1 \leq (1-\alpha)C_0$, where $\alpha < \beta$.

Trivial, we could select C_0 with this assumption.

Hypothesis: For all $l < k$ the (1) holds.

Step: From the hypothesis we have an upper bound $(1-\alpha)^{k-1}C_0 = \hat{h}_{k-1} \geq h_{k-1}$. Let us check if usage of it in (3) will give us $h_k \leq (1-\alpha)^k C_0$.

$$h_k \leq (1-\beta)h_{k-1} + \varepsilon_k(c_1 + c_2\sqrt{h_{k-1}}) \leq (1-\beta)\hat{h}_{k-1} + \frac{(\beta-\alpha)\hat{h}_{k-1}(c_1 + c_2\sqrt{h_{k-1}})}{c_1 + c_2\sqrt{\hat{h}_{k-1}}} \leq (1-\alpha)\hat{h}_{k-1} \leq (1-\alpha)^k C_0.$$

this concludes our proof.

1.2 GD with inexact Moreau-Envelope

Let us consider problem

$$\min_{x\in\mathbb{R}^n} f(x) + r(x), \tag{5}$$

where f is convex and L-smooth and r is convex, l.s.c. and nonsmooth. Then let us consider $\kappa > 0$ Moreau-Yosida envelope of this function

$$M_\kappa(y) = \min_{x\in\mathbb{R}^d} \left\{h_\kappa(x,y)\right\}, \tag{6}$$

where $h_\kappa(x,y) = f(x) + r(x) + \frac{\kappa}{2}\|x-y\|_2^2$. It has an important property that $x^\star$ is the unique minimizer of (5) iff it is the unique minimizer of

$$\min_{x\in\mathbb{R}^d} M_\kappa(x). \tag{7}$$

Moreover, if $f+g$ is convex and lower semicontinuous then M_κ is a smooth function with κ_l-Lipschitz continuous gradient

$$\nabla M_\kappa = \kappa(x - p_\kappa(x)), \tag{8}$$

where $p(x)$ the proximal point of x:

$$p_\kappa(x) = \operatorname*{argmin}_{y\in\mathbb{R}^n} \{h_\kappa(x,y)\}. \tag{9}$$

Then the GD algorithm for minimizing M_κ is the following:

$$x^{k+1} = x^k - \gamma\nabla M_\kappa(x^k) = x^k(1-\gamma\kappa) + \gamma\kappa p_\kappa(x^k). \tag{10}$$

Theorem 1 (Strongly convex case)**.** *Assume that* f *is* μ*-strongly convex. Choose* $\gamma \in (0, \frac{2}{\kappa+\frac{\mu\kappa}{\mu+\kappa}}]$*, then algorithm with updates* (10) *converges to the minimum with linear speed:*

$$\|x^{k+1}-x^\star\|_2^2 \le \left(1 - \frac{2\gamma\mu\kappa}{\mu\kappa+\kappa(\mu+\kappa)}\right)^{k+1}\|x^0-x^\star\|_2^2.$$

But in practice problem (9) has no analytical solution and requires for another optimization algorithm ti solve it. It brings inexactness in computation of $p_\kappa(x)$. Let us define inexact solution $p_\kappa^\varepsilon(x)$ as following:

$$\|p_\kappa^\varepsilon(x) - p_\kappa(x)\| < \frac{\varepsilon}{\kappa}. \tag{11}$$

It implies that the inexact gradient $\nabla^\varepsilon M_\kappa(x) = \kappa(x - p_\kappa^\varepsilon(x))$ is ε approximation of the real one

$$\|\nabla^\varepsilon M_\kappa(x) - \nabla M_\kappa(x)\| \le \varepsilon.$$

Let us now consider an algorithm with inexact gradient:

$$x^{k+1} = x^k - \gamma\nabla^\varepsilon M_\kappa(x^k) = x^k(1-\gamma\kappa) + \gamma\kappa p_\kappa^\varepsilon(x^k). \tag{12}$$

Theorem 2 (Strongly convex case)**.** *Assume that* f *is* μ*-strongly convex. Choose* $\alpha < beta$*,* $\gamma \in (0, \frac{2}{\kappa+\frac{\mu\kappa}{\mu+\kappa}}]$ *and sequence*

$$\varepsilon_k \le \frac{(\beta-\alpha)(1-\alpha)^{k-1}\|x^0-x^\star\|_2^2}{\gamma^2\varepsilon_k + \gamma(2\kappa\gamma+1)(1-\alpha)^{\frac{k-1}{2}}\|x^0-x^\star\|_2}, \tag{13}$$

where $\beta = \frac{2\gamma\mu\kappa}{\mu\kappa+\kappa(\mu+\kappa)}$ *Then algorithm with updates* (12) *converges to the minimum with linear speed:*

$$\|x^{k+1}-x^\star\|_2^2 \le (1-\alpha)^{k+1}\|x^0-x^\star\|_2^2.$$

Note that for simplicity instead of (16) cold be used the following bound:

$$\varepsilon_k \le \frac{(\beta-\alpha)(1-\alpha)^{\frac{k-1}{2}}\|x^0-x^\star\|_2}{\gamma(2\kappa\gamma+1)}. \tag{14}$$

Proof.

$$
\begin{aligned}
\|x^{k+1}-x^k\|_2^2 &= \|x^k-\gamma\nabla M_\kappa^{\varepsilon_k}(x^k)-x^\star\|_2^2 = \|x^k-x^\star\|_2^2+\gamma^2\|\nabla M_\kappa^{\varepsilon_k}(x^k)\|_2^2-2\gamma\langle\nabla M_\kappa^{\varepsilon_k}(x^k),x^k-x^\star\rangle\\
&= \|x^k-x^\star\|_2^2+\gamma^2\|\nabla M_\kappa^{\varepsilon_k}(x)-\nabla M_\kappa(x^\star)\|_2^2-2\gamma\langle\nabla M_\kappa^{\varepsilon_k}(x)-\nabla M_\kappa(x^\star),x^k-x^\star\rangle\\
&= \|x^k-x^\star\|_2^2+\gamma^2\left(\|\nabla M_\kappa(x)-\nabla M_\kappa(x^\star)\|_2^2+\|\nabla M_\kappa^{\varepsilon_k}(x^k)-\nabla M_\kappa(x^k)\|_2^2\right.\\
&\qquad\qquad\qquad\qquad\left.+2\langle\nabla M_\kappa(x)-\nabla M_\kappa(x^\star),\nabla M_\kappa^{\varepsilon_k}(x^k)-\nabla M_\kappa(x^k)\rangle\right)\\
&\qquad -2\gamma\left(\langle\nabla M_\kappa(x)-\nabla M_\kappa(x^\star),x^k-x^\star\rangle+\langle\nabla M_\kappa^{\varepsilon_k}(x^k)-\nabla M_\kappa(x^k),x^k-x^\star\rangle\right)\\
&\le \|x^k-x^\star\|_2^2+\gamma^2\|\nabla M_\kappa(x)-\nabla M_\kappa(x^\star)\|_2^2-2\gamma\langle\nabla M_\kappa(x)-\nabla M_\kappa(x^\star),x^k-x^\star\rangle\\
&\quad +\gamma^2\left(\varepsilon_k^2+\varepsilon_k\|\nabla M_\kappa(x)-\nabla M_\kappa(x^\star)\|_2\right)+2\gamma\varepsilon_k\|x^k-x^\star\|_2\\
&\le (1-\beta)\|x^k-x^\star\|_2^2+\gamma^2\varepsilon_k^2+(\gamma^2\kappa+2\gamma)\varepsilon_k\|x^k-x^\star\|_2.
\end{aligned}
$$

To conclude the proof we just need to use (3). □

1.3 Different κ

It is easy to see that we never use in proof that functions M_κ are the same as far as the analysis for every iterate is independent from all the precious ones. The only thing that we used is $\nabla M_\kappa(x^\star)=0$. It implies that for any sequence $\kappa_k>0$ an algorithm with update

$$x^{k+1}=x^k-\gamma\nabla^\varepsilon M_{\kappa_k}(x^k)=x^k(1-\gamma\kappa_k)+\gamma\kappa_k p^\varepsilon_{\kappa_k}(x^k). \tag{15}$$

Theorem 3 (Strongly convex case)**.** *Assume that* f *is* μ*-strongly convex. Choose* $\alpha_k<\beta_k$, $\gamma_k\in(0,\frac{2}{\kappa+\frac{\mu\kappa_k}{\mu+\kappa_k}}]$ *and sequence*

$$\varepsilon_k\le\frac{(\beta_k-\alpha_k)(1-\alpha_k)^{k-1}\|x^0-x^\star\|_2^2}{\gamma_k^2\varepsilon_k+\gamma_k(2\kappa_k\gamma_k+1)(1-\alpha_k)^{\frac{k-1}{2}}\|x^0-x^\star\|_2}, \tag{16}$$

where $\beta_k=\frac{2\gamma_k\mu\kappa_k}{\mu\kappa_k+\kappa_k(\mu+\kappa_k)}$ *Then algorithm with updates* (15) *converges to the minimum with the linear speed:*

$$\|x^{k+1}-x^\star\|_2^2\le\prod_{l=1}^{k}(1-\alpha_l)\|x^0-x^\star\|_2^2.$$

1.4 κ selection

2 *Catalyst with SPY*

In contrast with the previous section let us consider now an accelerated version of algorithm (12). It is exactly Catalyst algorithm.

Algorithm 1 Catalyst

Input: $x_0 \in \mathbb{R}^n$, smoothing parameter κ, optimization method $\mathcal{M}$, $y_0 = x_0$, $q = \frac{\mu}{\mu+\kappa}$
Output: $x^\star \in \operatorname{argmin}_{x\in\mathbb{R}^n} F(x)$
while *desired stopping criterion is not satisfied* **do**

Find x_k using $\mathcal{M}$

$$x_k \in_{x\in\mathbb{R}^n} \{H_k(x) \triangleq F(x) + \frac{\kappa}{2}\|x - y_{k-1}\|^2\} \tag{17}$$

Compute $\alpha_k \in (0;1)$ from $\alpha_k{}^2 = (1-\alpha_k)\alpha_{k-1}{}^2 + q\alpha_k$ Compute y_k using β_k from $(0,1)$

$$y_k = x_k + \beta_k(x_k - x_{k-1}), \tag{18}$$

where

$$\beta_k = \frac{\alpha_{k-1}(1-\alpha_{k-1})}{\alpha_{k-1}{}^2 + \alpha_k}$$

end

Consider the strongly convex objective f then the following theoretical result makes sense.

Lemma 1 (Theorems 3.1 and 3.3 [?]). *When $\mu = 0$, choose $\alpha_0 = (\sqrt{5}-1)/2$ and*

$$\varepsilon_k = \frac{2(H_k(x_0) - H_k^\star)}{9(k+2)^{4+\eta}} \quad \text{with} \quad \eta > 0. \tag{19}$$

Then Algorithm 1 generates iterates $(x_k)_k$ such that

$$F(x_k) - F^\star \le \frac{8}{(k+2)^2}\left(\left(1+\frac{2}{\eta}\right)^2 (F(x_0) - F^\star) + \frac{\kappa}{2}\|x_0 - x^\star\|^2\right). \tag{20}$$

If $\mu > 0$, choose $\alpha_0 = \sqrt{q}$ with $q = \mu/(\mu+\kappa)$ and

$$\varepsilon_k = \frac{2}{9}(H_k(x_0) - H_k^\star)(1-\rho)^k \quad \text{with} \quad \rho \le \sqrt{q}. \tag{21}$$

Then Algorithm 1 generates iterates $(x_k)_k$ such that

$$F(x_k) - F^\star \le C(1-\rho)^k\,(F(x_0) - F^\star) \quad \text{with} \quad C = \frac{8}{\sqrt{q}-\rho}. \tag{22}$$

3 κ bound for SPY

Consider the problem

$$\min_{x\in\mathbb{R}^n} F(x) = \sum_{i=1}^m \pi_i f_i(x) + r(x) \tag{23}$$

Let us thed define $h_{k,i} = f_i + r + \frac{\kappa}{2}\|x - y_{k-1}\|_2^2$.

Let us now calculate the parameter κ that makes problem $h_{k,i}$ as well-conditioned as required in the theorem. If function f_i is $L-$smooth and $\mu-$strongly convex (may be with $\mu = 0$), then problem $h_{k,i}$ is $(L+\kappa)-$smooth and $(\mu+\kappa)-$strongly convex. Then constant (with optimal/maximal stepsize) is the following:

$$\begin{aligned}
\alpha_k = \alpha_{k,i} = \frac{4(L+\kappa)(\mu+\kappa)}{(\mu+L+2\kappa)^2} &\geq p^{\max} + \beta - p^{\min} \\
&\Uparrow \text{ if } \mu \neq L \\
\frac{\mu+\kappa}{L+\kappa} &\geq p^{\max} - p^{\min} \\
&\Updownarrow \\
\kappa &\geq \frac{(p^{\max}-p^{\min})L - p^{\max}\mu}{p^{\min}}.
\end{aligned} \tag{24}$$

Note that if $p^{\max} = p^{\min}$ this bound boils down to the minimal κ such that inner problem is convex.

Is this κ the one that we need to select? On the one hand, bigger κ implies better conditioning of inner problem and as a result faster convergence. On the other hand, the amount of restarts needed grows with increasing κ. Let us present an "optimal" value, that takes into account both this aspects.

3.1 Adaptive $\mathbf{S}^k$

Let us clarify our specific $\mathbf{S}^k$ selection for ℓ_1 regularized problems.

Assumption 1. *The sparsity mask selectors* $(\mathbf{S}^k)$ *are random variables such that* $\mathbb{P}[j \in \mathbf{S}^k] = 1$ *if* $j \in \operatorname{supp}(x^k)$ *and* $\mathbb{P}[j' \in \mathbf{S}^k] = p > 0$ *for all* $j' \notin \operatorname{supp}(x^k)$.

3.2 Communication metric

In the epoch of large-scale data in the algorithm complexity it's important to take into account "communication time" that is really "size-dependent". According to this, let's consider as a "communications' metric" the total amount of data sent (in both ways from and to master). Let's assume that the moment of identification already took a place (we could assume this as far as this moment is a finite one). Then, after this, both algorithms has the same structure of gain for inner loop "iteration" $(1 - \mu_F/L_F)$, where $\mu_F = \mu + \kappa$ and $L_F = L + \kappa$ that does not depend on p in adaptive case also, but they have different amount of exchanges (assuming that epochs that come from delays have the same structure in both algorithms) $s + n$ Vs $2s + p(n - s)$ in DAve Vs SPY-DR correspondingly. In the same time the κ for adapted version is not the optimal one, that makes it worse in terms of iterations.

Let us present the way to select probability parameter p such a way that sparsified algorithm would be better than the full one.

Theorem 4. *Let* ϱ *be the sparsity of the final solution* $|\operatorname{supp}(x^\star)| = \varrho n$. *Choose* $p = \frac{2\varrho}{3\varrho+1}$ *then Algorithm* **??** *converges* $\tilde{O}\left(\sqrt{\frac{1+\varrho}{2\varrho}}\right)$ *faster than without sparsification in terms of communications made.*

Proof. Taking into account the finiteness of identification time for both algorithms we consider the moment, when identification happens. In other words we assume that total size of communication round is $n(1+\varrho)$ for nonsparsified algorithm and $n(2\varrho + p(1-\varrho))$ for sparsified with parameter p. Let us first present the communication complexity of nonsparsified algorithm.

$$\tilde{O}\left(\frac{L+\kappa}{\sqrt{(\mu+\kappa)\mu}} \log\frac{1}{\varepsilon} n(\varrho+1)\right) = \tilde{O}\left(\frac{L+L-2\mu}{\sqrt{(\mu+L-2\mu)\mu}} \log\frac{1}{\varepsilon} n(\varrho+1)\right) = \tilde{O}\left(\frac{2\sqrt{L-\mu}}{\sqrt{\mu}} \log\frac{1}{\varepsilon} n(\varrho+1)\right).$$

Let us now calculate κ taking into account the proposed $p = \frac{2\varrho}{3\varrho+1}$:

$$\kappa = \frac{(1-p)L - \mu}{p} = \frac{\left(1 - \frac{2\varrho}{3\varrho+1}\right)L - \mu}{\frac{2\varrho}{3\varrho+1}} = \frac{(\varrho+1)L - (3\varrho+1)\mu}{2\varrho}.$$

Then the communication complexity of p-sparsified algorithm is

$$\tilde{O}\left(\frac{L+\kappa}{\sqrt{(\mu+\kappa)\mu}}\log\frac{1}{\varepsilon}n(2\varrho+p(1-\varrho))\right)=\tilde{O}\left(\frac{\frac{3\varrho+1}{2\varrho}(L-\mu)}{\sqrt{\frac{\varrho+1}{2\varrho}(L-\mu)\mu}}\log\frac{1}{\varepsilon}n\left(2\varrho+\frac{2\rho(1-\varrho)}{3\varrho+1}\right)\right)$$
$$=\tilde{O}\left(\frac{4(\varrho^2+\varrho)\sqrt{L-\mu}}{\sqrt{(\varrho+1)2\varrho\mu}}\log\frac{1}{\varepsilon}n\right).$$

To finish the comparison the last thing to compare is

$$\frac{4(\varrho^2+\varrho)}{\sqrt{(\varrho+1)2\varrho}}\leq 2\varrho+2 \Leftarrow \sqrt{2\varrho}\leq\sqrt{\varrho+1}\Leftarrow 0\leq\varrho\leq 1.$$

□

It is important now to present some remarks on this result. First, there is no dependence on problem conditioning for both: gain and probability selection. Second, $p\to 0$ when $\varrho\to 0$, and $p\to 0.5$ when $\varrho\to 1$. The last could be explained with the way of selecting κ, such that $\kappa(p)\xrightarrow{p\to 0.5}\kappa^\star$. Finally, p depends on the unknown sparsity of the final solution, so to use such probability starting from some moment an adaptive probability selection should be used. That implies adaptive κ selection in Catalyst. if we forget about adaptive catalyst and consider the one with fixed κ will we have a gain?

Let us check if we could have a profit if we don't know the final sparsity ϱ. First, consider $p=0.5$, then

$$\tilde{O}\left(\frac{L+\kappa}{\sqrt{(\mu+\kappa)\mu}}\log\frac{1}{\varepsilon}n(2\varrho+p(1-\varrho))\right)=\tilde{O}\left(\frac{L+\kappa}{\sqrt{(\mu+\kappa)\mu}}\log\frac{1}{\varepsilon}(0.5n+1.5\varrho)\right)$$

that is always smaller than the full update for any sparsity and reaches the 2 times faster speed if $\varrho\ll 1$. (note that $p=0.5$ implies bound on $\kappa>L-2\mu$ that is an optimal one).