Journal Club

A Universal Catalyst for First-Order Optimization
(H. Lin, J. Mairal and Z. Harchaoui)

CMAP, Ecole Polytechnique

March 8th, 2018

Plan

1. Motivations
2. Existing Acceleration Methods
3. Universal Catalyst
4. Conclusion

Motivations

- Unconstrained Minimization of a large sum of functions

$$\min_{x \in \mathbb{R}^p} \left\{ F(x) \triangleq \frac{1}{n} \sum_{i=1}^{n} f_i(x) + \psi(x) \right\} \tag{1}$$

where f_i are L_i-smooth and convex and ψ is a convex penalty but not necessarily differentiable.

$$|\nabla f_i(x) - \nabla f_i(y)| < L_i |x - y| \tag{2}$$

- **Goal:** Provide an acceleration scheme that can apply to existing un-accelerated methods
- Acceleration in the sense of Nesterov

- Given training data $(y_i, z_i)_{i=1}^n$ where y_i are responses and z_i are regressors. x here represents the model parameters.
- F is the loss function and measures how well the model fit the training data and ψ prevents from overfitting
- Example: logistic regression. Reponses y_i take values in $\{0, 1\}$:

$$\min_{x \in \mathbb{R}^p} \left\{ \frac{1}{n} \sum_{i=1}^{n} \log(1 + e^{-y_i < x, z_i >}) + \lambda \|x\|^2 \right\} \tag{3}$$

- Large-scale dimension leads to first-order gradient-based methods

- Generic acceleration scheme, which applies to previously unaccelerated algorithms such as SVRG, SAG, SAGA, SDCA, MISO, or Finito, and which is not tailored to finite sums.
- Complexity analysis for μ-strongly convex objectives.
- Complexity analysis for non-strongly convex objectives.

Existing Acceleration Methods

- Classical way to solve the problem without the penalty ($\min_{x \in \mathbb{R}^p} f(x)$) is by gradient descent method (L smooth objective function):

$$x^k = x^{k-1} - \frac{1}{L}\nabla f(x^{k-1}) \tag{4}$$

- Can be viewed as a proximal regularisation of the linearized function f at x^{k-1} (Beck, Teboulle, 2009):

$$x^k = \arg\min_{x \in \mathbb{R}^p} \{f(x^{k-1}) + \langle x - x^{k-1}, \nabla f(x^{k-1})\rangle + \frac{1}{L}\|x - x^{k-1}\|^2\} \tag{5}$$

- Leads to ISTA when adding a penalty

$$x^k = \arg\min_{x \in \mathbb{R}^p} \frac{1}{2}\{\|x - (x^{k-1} - \frac{1}{L}\nabla f(x^{k-1}))\|^2 + \frac{1}{L}\psi(x)\} \tag{6}$$

- (1980), Nesterov introduced an acceleration scheme adding a memory term to the descent:

$$x^k = \arg\min_{x \in \mathbb{R}^p} \frac{1}{2}\{\|x - (y^{k-1} - \frac{1}{L}\nabla f(y^{k-1}))\|^2 + \frac{1}{L}\psi(x)\} \quad (7)$$

with $y^{k-1} = x^{k-1} + \beta^k(x^{k-1} - x^{k-2})$ and $0 < \beta^k < 1$

- Complexity to reach an $\epsilon-$ solution:

Algo	$\mu > 0$	$\mu = 0$
ISTA	$\mathcal{O}(n\frac{L}{\mu}\log(1/\epsilon))$	$\mathcal{O}(\frac{nL}{\epsilon})$
FISTA	$\mathcal{O}(n\sqrt{\frac{L}{\mu}}\log(1/\epsilon))$	$\mathcal{O}(\frac{nL}{\sqrt{\epsilon}})$

- $\epsilon-$solution means $f(x^k) - f(x^*) \leq \epsilon$
- Large sum structure of f not exploited here

- Randomized algorithms take into account the structure of the objective function and compute only one random gradient at each iteration which yields a better **expected** computation complexity
- To get $\mathbb{E}\left[f(x^k) - f(x^*)\right] \leq \epsilon$ we need $\mathcal{O}(1/\epsilon)$ iterations

Algo	$\mu > 0$
SAG, SAGA, MISO etc..	$\mathcal{O}(\max\left(n, \frac{L}{\mu}\right)\log(1/\epsilon))$
FISTA	$\mathcal{O}(n\sqrt{\frac{L}{\mu}}\log(1/\epsilon))$

- Acceleration when the number of observations is large enough:

$$\max\left(n, \frac{L}{\mu}\right) \leq n\sqrt{\frac{L}{\mu}} \Rightarrow n \geq \sqrt{\frac{L}{\mu}} \tag{8}$$

- Not in the sense of Nesterov though (Acceleration due to incremental update, not to a memory term)
- See Bottou et. al. 2018

Universal Catalyst

- Challenge: can we accelerate these algorithms by a universal scheme for both convex and strongly convex objectives ?
- Given any algorithm $\mathcal{M}$ that can solve a convex problem, at iteration k, rather than minimizing $F(x)$, use as many iterations of $\mathcal{M}$ as needed to minimize:

$$G^k(x) \triangleq F(x) + \frac{\mathcal{K}}{2}\|x - y^{k-1}\|^2 \tag{9}$$

  such that $G^k(x) - G^* \leq \epsilon^k$.
- Compute $y^k = x^k + \beta^k(x^k - x^{k-1})$ with $\beta^k = \frac{\alpha_{k-1}(1-\alpha_{k-1})}{\alpha_{k-1}^2+\alpha_k}$, $\alpha_k^2 = (1-\alpha_k)\alpha_{k-1}^2 + q\alpha_k$ and $q = \frac{\mu}{\mu+\mathcal{K}}$
- The Catalyst algorithm $\mathcal{A}$ is a wrapper of $\mathcal{M}$ that takes advantage of both basic M-M scheme and Nesterov acceleration

- G^k is easier to minimize than F
  - G^k is always strongly convex as long as F is convex
  - G^k has a better condition number when F is strongly convex $(\frac{L+\mathcal{K}}{\mu+\mathcal{K}} < \frac{L}{\mu})$
- need to find a trade-off between $\mathcal{K} >> 1$ (easy) and $\mathcal{K} = 0$.
- Inner loop: How many iterations of $\mathcal{M}$ to obtain the ϵ^k precision $(G^k(x) - G^* \leq \epsilon^k)$
- Outter loop: with the sequences of (x^k) obtained by $\mathcal{M}$, wisely choose the update y^k (stepsize β^k) to obtain optimal rate on $F(x^k) - F^*$

Main Theorem For strongly convex objective

- Choose $\alpha_0 = \sqrt{q}$ and $q = \frac{\mu}{\mu + \mathcal{K}}$ and the sequence:

$$\epsilon^k = \frac{2}{9}(F(x^0) - F^*)(1-\rho)^k \tag{10}$$

- Then the algorithm generates iterates (x^k) such that:

$$F(x^k) - F^* \leq C(1-\rho)^{k+1}(F(x^0) - F^*) \quad \text{with} \quad C = \frac{8}{(\sqrt{q}-\rho)^2} \tag{11}$$

- In practice $\rho = 0.9\sqrt{q}$ and since we don't know F^* for non negative function we can set $\epsilon^k = \frac{2}{9}F(x^0)(1-\rho)^k$

- Choose $\alpha_0 = (\sqrt{5}-1)/2$ and the sequence:

$$\epsilon^k = \frac{2(F(x^0) - F^*)}{9(k+2)^{4+\eta)}} \tag{12}$$

- Then the algorithm generates iterates (x^k) such that:

$$F(x^k) - F^* \leq \frac{8}{(k+2)^2}((1+2/\eta)^2 F(x^0) - F^* + \frac{\mathcal{K}}{2}\|x^0 - x^*\|^2) \tag{13}$$

- In practice $\eta = 0.1$

- An appropriate $\mathcal{M}$ (applied to G^k) for a strongly convex objective function HAS to ta have a linear convergence rate, i.e. there exists $\tau_{\mathcal{M}}$ such that:

$$G^k(z^t) - G^{k*} \leq (1 - \tau_{\mathcal{M}})^t (G^k(z^0) - G^{k*}) \tag{14}$$

- $\tau_{\mathcal{M}}$ depends on the condition number. ISTA: $\tau_{\mathcal{M},F} = \mu/L$ and FISTA: $\tau_{\mathcal{M},F} = \sqrt{\mu/L}$
- Thanks to the quadratic term added to F we can achieve faster rates since $\tau_{\mathcal{M},G^k} = \frac{\mu+\mathcal{K}}{L+\mathcal{K}} > \tau_{\mathcal{M},F}$
- With the proposed sequence (ϵ^k) the precision is reached, choosing $z^0 = x^{k-1}$ with
    - Strongly convex case: constant number of iterations $\tilde{\mathcal{O}}(\frac{1}{\tau_{\mathcal{M}}})$
    - Convex case: constant number of iterations $\tilde{\mathcal{O}}(\frac{1}{\tau_{\mathcal{M}}}) \log(k+2)$

Conclusion

Case when $n \leq L/\mu$ when $\mu > 0$

Algo	$\mu > 0$	$\mu = 0$	Cat. $\mu > 0$	Cat. $\mu = 0$
FG	$\mathcal{O}(n\frac{L}{\mu}\log(\frac{1}{\epsilon}))$	$\mathcal{O}(n\frac{L}{\epsilon})$	$\tilde{\mathcal{O}}(n\sqrt{\frac{L}{\mu}}\log(\frac{1}{\epsilon}))$	$\tilde{\mathcal{O}}(n\frac{L}{\sqrt{\epsilon}})$
SAGA	$\mathcal{O}(\frac{L}{\mu}\log(\frac{1}{\epsilon}))$	$\mathcal{O}(n\frac{L}{\epsilon})$	$\tilde{\mathcal{O}}(\sqrt{\frac{nL}{\mu}}\log(\frac{1}{\epsilon}))$	$\tilde{\mathcal{O}}(n\frac{L}{\sqrt{\epsilon}})$
MISO	$\mathcal{O}(\frac{L}{\mu}\log(\frac{1}{\epsilon}))$	NA	$\tilde{\mathcal{O}}(\sqrt{\frac{nL}{\mu}}\log(\frac{1}{\epsilon}))$	$\tilde{\mathcal{O}}(n\frac{L}{\sqrt{\epsilon}})$

- Plus:
  - Simple acceleration scheme that applies to large class of methods
  - Recover Optimal rates for known algorithms
  - Simple to implement
- Minus:
  - Acceleration when $n \leq L/\mu$ otherwise hard to beat $0(n\log(1/\epsilon))$
  - μ is just an estimate of the true strong convexity $\mu' \geq \mu$
  - When $n \leq L/\mu$ but $n \geq L/\mu'$ apperas to be hard to accelerate.

Thank you