A Superlinearly-Convergent Proximal Newton-Type Method for the Optimization of Finite Sums

Anton Rodomanov — anton.rodomanov@gmail.com
Dmitry Kropotov — dmitry.kropotov@gmail.com

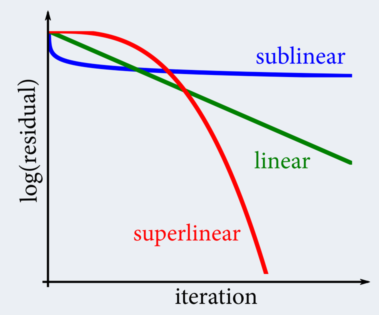
p(B|A)yesgroup.ru

Motivation

- Consider the minimization of the composite finite-average of many functions:

$$\min_x \left[\phi(x) := \frac{1}{n}\sum_{i=1}^n f_i(x) + h(x) \right],$$

  where f_i are twice continuously differentiable and convex, h is closed convex.
- Big data setting: n is very large (millions, billions etc.).
- Incremental/stochastic optimization methods, which process only one f_i at each iteration, are among the most effective methods for this task.
- There exists many different incremental optimization schemes:
  - SGD, oLBFGS [Schraudolph et al., 2007], AdaGrad [Duchi et al., 2011], SQN [Byrd et al., 2014], Adam [Kingma, 2014] etc.
  - SAG [Schmidt et al., 2013], SVRG [Johnson & Zhang, 2013], SAGA [Defazio et al., 2014a], MISO [Mairal, 2015] etc.
- They all have either a sublinear or linear convergence rate.
- Goal: an incremental optimization method with a superlinear rate of convergence.



Main idea

- Build the second-order Taylor approximation of each f_i:

$$m_k^i(x) := f_i(v_k^i) + \nabla f_i(v_k^i)^\top (x - v_k^i) + \frac{1}{2}(x - v_k^i)^\top \nabla^2 f_i(v_k^i)(x - v_k^i).$$

- Then ϕ can be approximated with $m_k(x) := \frac{1}{n}\sum_{i=1}^n m_k^i(x) + h(x)$.
- Find the minimizer of the model: $\bar{x}_k := \operatorname{argmin}_x m_k(x)$.
- Choose next iterate x_{k+1} between x_k and $\bar{x}_k$: $x_{k+1} = x_k + \alpha_k(\bar{x}_k - x_k)$.
- Each time update only one v_k^i to keep the iteration cost independent of n:

$$v_{k+1}^i := \begin{cases} x_{k+1} & \text{if } i = i_k, \\ v_k^i & \text{otherwise,} \end{cases}$$

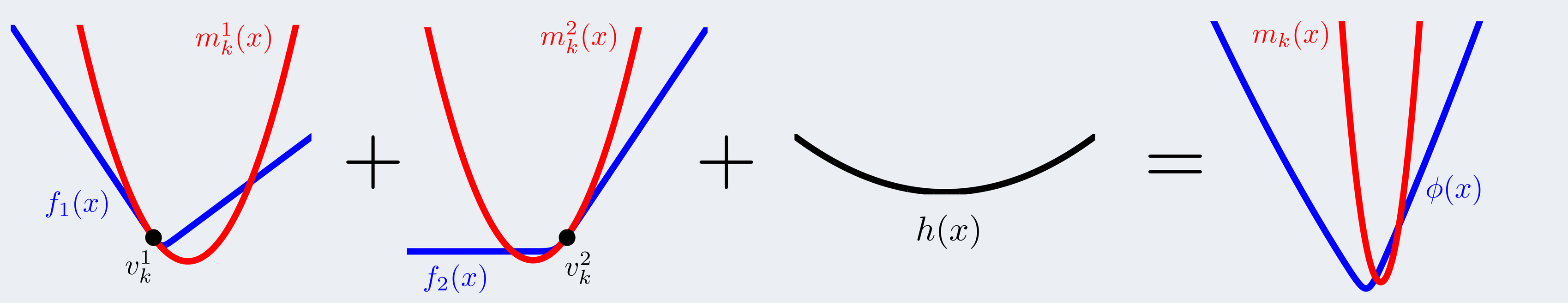
  where $i_k \in \{1, \ldots, n\}$ is the index of the component to update.
- Note: m_k is a (composite) quadratic,

$$m_k(x) = (g_k - u_k)^\top x + \frac{1}{2} x^\top H_k x + h(x) + \text{const},$$

  and is determined only by the following three quantities:

$$H_k := \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(v_k^i), \quad u_k := \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(v_k^i) v_k^i, \quad g_k := \frac{1}{n}\sum_{i=1}^n \nabla f_i(v_k^i)$$

  which can be updated in iterations using the "add-subtract" principle.



Inexact model minimization

- In general, there is no need to find the minimizer $\bar{x}_k$ of the model exactly.
- Define the composite gradient mapping:

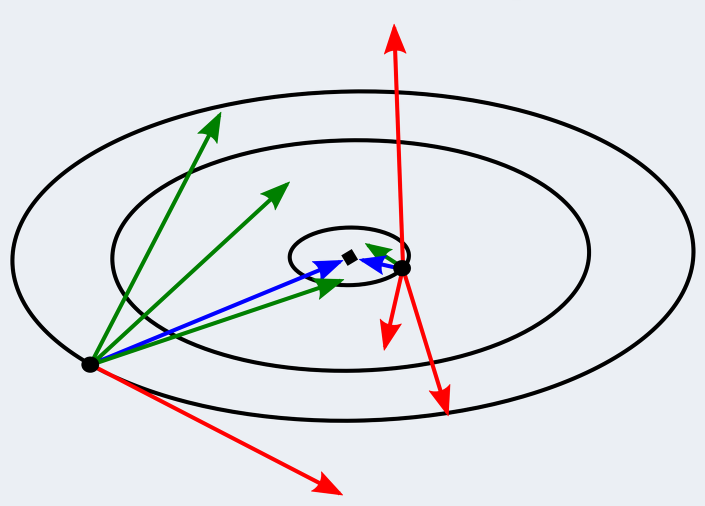
$$T_L(x, \xi) := \operatorname*{argmin}_y \left[\xi^\top y + \frac{L}{2}\|y - x\|^2 + h(y) \right],$$

$$G_L(x, \xi) := L(x - T_L(x, \xi)).$$

  Note: for $h \equiv 0$, we have $G_L(x, \xi) \equiv \xi$.
- We show that, instead of $\bar{x}_k = \operatorname{argmin}_x [m_k(x) =: s_k(x) + h(x)]$, any point $\hat{x}_k = T_L(x; \nabla s_k(x))$ can be used in NIM provided that

$$\|G_L(x, \nabla s_k(x))\| \le \min\{1, (\Delta_k)^\gamma\}\Delta_k, \qquad \Delta_k := \|G_1(\bar{v}_k, g_k)\|.$$

  Here L can be any such that $L \ge L_0 \equiv 1$, $\bar{v}_k := \frac{1}{n}\sum_{i=1}^n v_k^i$ and $\gamma \in (0, 1]$.
- Intuition: the closer NIM to the optimum, the more accurate $\hat{x}_k$ is required.
- Possible inner solver: Fast Gradient Method [Nesterov, 2013].



Algorithm NIM (Newton-type Incremental Method)

1: **Input:** $x_0, \ldots, x_{n-1} \in \mathbb{R}^d$: initial points; $\{\alpha_k\}$: step lengths.
2: Initialize model: $v^i := x_{i-1}$ for $i = 1, \ldots, n$ and
3: $H := \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(v^i)$, $u := \frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(v^i) v^i$, $g := \frac{1}{n}\sum_{i=1}^n \nabla f_i(v^i)$
4: **for** $k \ge n - 1$ **do**
5: Compute minimizer: $\hat{x} \approx \operatorname{argmin}_x \left[(g - u)^\top x + \frac{1}{2}x^\top H x + h(x)\right]$
6: Make a step: $x_{k+1} := x_k + \alpha_k(\hat{x} - x_k)$
7: Update model for $i := (k+1) \bmod n + 1$ (cyclic order):
8: $H := H + \frac{1}{n}\left[\nabla^2 f_i(x_{k+1}) - \nabla^2 f_i(v^i)\right]$
9: $u := u + \frac{1}{n}\left[\nabla^2 f_i(x_{k+1}) x_{k+1} - \nabla^2 f_i(v^i) v^i\right]$
10: $g := g + \frac{1}{n}\left[\nabla f_i(x_{k+1}) - \nabla f_i(v^i)\right]$
11: $v^i = x_{k+1}$
12: **end for**

Convergence rate

- Suppose ∇f_i and $\nabla^2 f_i$ are Lipschitz-continuous with constants L_f and M_f.
- Assume x^* is a minimizer of ϕ with $\frac{1}{n}\sum_{i=1}^n \nabla^2 f_i(x^*) \succeq \mu_f I \succ 0$, and all the initial points are close enough to x^*: $\|x_i - x^*\| \le R$ for $0 \le i \le n - 1$.
- Then the sequence of iterates $\{x_k\}$ of NIM with $\alpha_k \equiv 1$ converges to x^* at an R-superlinear rate, i.e. there exist $\{z_k\}$ and $\{q_k\}$ such that for $k \ge n$

$$\|x_k - x^*\| \le z_k, \qquad z_{k+1} \le q_k z_k, \qquad q_k \to 0.$$

  If the model is minimized exactly, i.e. $\hat{x}_k = \bar{x}_k$, then

$$R := \frac{\mu_f}{2M_f}, \qquad q_k := \left(1 - \frac{3}{4n}\right)^{2^{\lceil k/n \rceil - 1}}.$$

  If the model is minimized inexactly using the proposed conditions, then

$$R := \min\left\{\frac{\mu_f}{2M_f}, \left(\frac{\mu_f^3}{128(2 + L_f)^{5+2\gamma}}\right)^{1/(2\gamma)}\right\}, \qquad q_k := \left(1 - \frac{7}{16n}\right)^{(1+\gamma)^{\lceil k/n \rceil}/2}.$$

- For certain types of h (e.g. when h is differentiable or an indicator function) one can prove a global linear convergence of NIM for a small enough step size.

Order of component selection (cyclic vs randomized)

- Consider $f_1(x) := \frac{1}{2}\|x\|^2 + \frac{n}{3}\|x\|^3$, $f_i(x) := \frac{1}{3}\|x\|^3$, $i > 1$, $h \equiv 0$.
- If one uses $i \sim \operatorname{Unif}\{1, \ldots, n\}$ in NIM and $\|x_0 - x^*\| < 1$, then

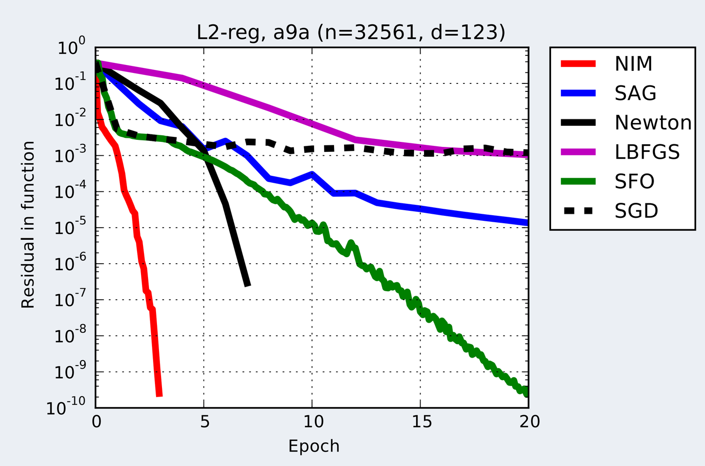
$$\mathbb{E}[\|x_k - x^*\|^2] \ge \frac{1}{3}\left(1 - \frac{1}{n}\right)^{k-n}\|x_0 - x^*\|^2,$$

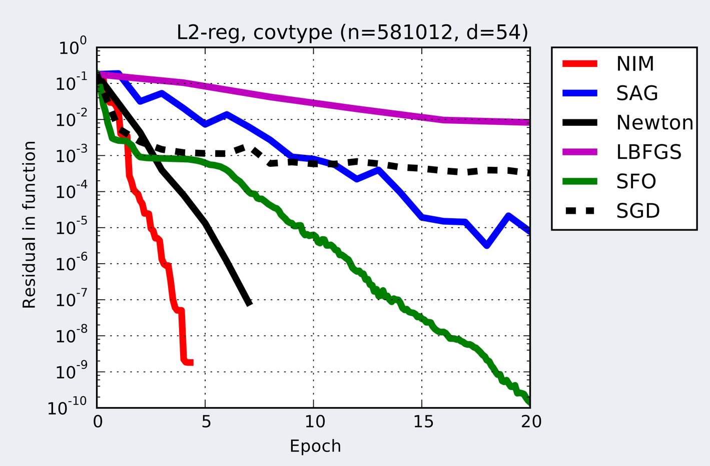
  which is a linear convergence rate.
- At the same time, for $i = (k+1) \bmod n + 1$ and $k \ge n$, we get

$$\|x_k - x^*\| \le \|x_{k-n} - x^*\|^2.$$

  i.e. a quadratic rate w.r.t. epochs and superlinear rate w.r.t. iterations.

Experiments (logistic regression)





L2-reg	SUSY (n=5 000 000, d=18)				alpha (n=500 000, d=500)				mnist8m (n=8 100 000, d=784)			
Res	NIM	SAG	Newton	LBFGS	NIM	SAG	Newton	LBFGS	NIM	SAG	Newton	LBFGS
10^{-1}	**.09s**	2.71s	2.48s	1.78s	1.91s	**1.36s**	1.6m	4.01s	57.68s	**34.91s**	47.8m	1.1m
10^{-2}	**.13s**	3.84s	4.30s	2.52s	13.37s	**6.72s**	2.6m	17.68s	**1.6m**	2.1m	1.4h	5.2m
10^{-4}	**1.36s**	1.3m	11.33s	2.60s	36.65s	**36.04s**	3.4m	58.35s	16.7m	**7.1m**	-	1.6h
10^{-5}	**2.78s**	1.9m	14.43s	4.09s	**46.66s**	1.0m	3.6m	1.4m	**26.7m**	1.0h	-	-
10^{-6}	**3.95s**	2.2m	16.71s	5.26s	**53.92s**	1.5m	4.0m	1.9m	**33.5m**	-	-	-
10^{-8}	**5.30s**	2.6m	19.41s	8.43s	**1.0m**	2.7m	4.1m	2.8m	**46.0m**	-	-	-
10^{-10}	**5.95s**	3.4m	20.80s	9.01s	**1.2m**	4.3m	4.7m	3.4m	**53.3m**	-	-	-

L1-reg	SUSY (n=5 000 000, d=18)			alpha (n=500 000, d=500)			mnist8m (n=8 100 000, d=784)		
Res	NIM	SAG	Newton	NIM	SAG	Newton	NIM	SAG	Newton
10^{-1}	**.09s**	4.63s	2.72s	26.76s	**1.31s**	1.1m	15.7m	**33.62s**	53.6m
10^{-2}	**.89s**	6.55s	5.63s	44.94s	**6.52s**	1.8m	37.0m	**2.1m**	1.8h
10^{-4}	**3.31s**	-	13.12s	1.1m	**35.51s**	2.5m	1.0h	**7.3m**	3.1h
10^{-5}	**4.57s**	-	15.87s	1.3m	**1.0m**	2.9m	**1.2h**	1.4h	-
10^{-6}	**6.25s**	-	18.46s	**1.3m**	1.5m	3.1m	**1.5h**	-	-
10^{-8}	**11.56s**	-	38.31s	**1.5m**	2.9m	3.5m	**2.3h**	-	-
10^{-10}	**17.51s**	-	45.08s	**1.6m**	4.8m	4.5m	**3.4h**	-	-