جامعة الملك عبدالله للعلوم والتقنية
King Abdullah University of Science and Technology

One Method to Rule Them All:
Variance Reduction for Data, Parameters and Many New Methods

Filip Hanzely[1] Peter Richtárik[1, 2]

[1]KAUST [2]MIPT

MIPT MOSCOW INSTITUTE OF PHYSICS AND TECHNOLOGY

Optimization Problem

$$\min_{x\in\mathbb{R}^d} \frac{1}{n}\sum_{j=1}^n f_j(x) + \psi(x),$$

- $f_j : \mathbb{R}^d \to \mathbb{R}$ is M_j smooth and convex:
$$0 \preceq \nabla^2 f_j(x) \preceq M_j$$
- $\psi : \mathbb{R}^d \to \mathbb{R}\cup\{+\infty\}$ is a proper, closed and convex regularizer, admitting a cheap proximal operator
- $f \overset{\text{def}}{=} \frac{1}{n}\sum_j f_j$ is σ quasi strongly convex

Oracle

$G(x) \overset{\text{def}}{=} [\nabla f_1(x), \nabla f_2(x), \dots, \nabla f_n(x)]$: Jacobian matrix

- Oracle can be accessed via: $\mathcal{U}G(x)$, $\mathcal{S}G(x)$
- $\mathcal{U} : \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n}$ - random linear operator, identity in expectation
- $\mathcal{S} : \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n}$ - random projection operator, possibly correlated with $\mathcal{U}$
- $\mathcal{U}$, $\mathcal{S}$ might correspond to **right matrix multiplication** (`SAGA [1], JacSketch [2]`), **left matrix multiplication** (`SEGA` [3]), their **combination** (`ISAEGA`) and **many more**
- Different choices of $\mathcal{U}$, $\mathcal{S}$ yield different methods.

Variance reduction (unbiased)

Given sequence J^k which estimates $G(x^k)$ such that $\lim_{k\to\infty} J^k = G(x^*)$, unbiased variance reduced gradient is the following:

$$g^k = \frac{1}{n}J^k e + \frac{1}{n}\mathcal{U}\left(G(x^k) - J^k\right) e. \tag{1}$$

Jacobian Sketching

Observing $\mathcal{S}G(x^k)$ every iteration, how to design Jacobian estimator sequence J^k? Projecting:

$$\begin{aligned} J^{k+1} &= \operatorname{argmin}_J \|J - J^k\| \quad \text{s. t. } \mathcal{S}J = \mathcal{S}G(x^k) \\ &= J^k - \mathcal{S}(G(x^k) - J^k) \end{aligned} \tag{2}$$

Algorithm

Algorithm 1 Generalized `JacSketch` (`GJS`)

Parameters: Stepsize $\alpha > 0$, random projector $\mathcal{S}$ and unbiased sketch $\mathcal{U}$
Initialization: Choose solution estimate $x^0 \in \mathbb{R}^d$ and Jacobian estimate $J^0 \in \mathbb{R}^{d\times n}$
for $k = 0, 1, \dots$ **do**
  Sample realizations of $\mathcal{S}$ and $\mathcal{U}$, and perform sketches $\mathcal{S}G(x^k)$ and $\mathcal{U}G(x^k)$
  $J^{k+1} = J^k - \mathcal{S}(J^k - G(x^k))$ — update the Jacobian estimate via (2)
  $g^k = \frac{1}{n}J^k e + \frac{1}{n}\mathcal{U}\left(G(x^k) - J^k\right) e$ — construct the gradient estimator via (1)
  $x^{k+1} = \operatorname{prox}_{\alpha\psi}(x^k - \alpha g^k)$ — perform the proximal SGD step
end for

Convergence rate

Single convergence theorem, **tightest known rate in every special case** (many new rates in special cases for known methods; many new methods as well).

- Let $\mathcal{M} : \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n}$ be linear operator such that $(\mathcal{M}X)_{:j} = M_j X_{:j}$ for any $X \in \mathbb{R}^{d\times n}$.
- Let $\mathcal{B} : \mathbb{R}^{d\times n} \to \mathbb{R}^{d\times n}$ be a linear operator (to be chosen; only for theory) such that with stepsize α we have:

$$\begin{aligned} (1-\alpha\sigma)\left\|\mathcal{B}\mathcal{M}^{\dagger\frac{1}{2}}X\right\|^2 &\ge \frac{2\alpha}{n^2}\mathbb{E}\left[\|\mathcal{U}Xe\|^2\right] \\ &\quad + \left\|(\mathcal{I} - \mathbb{E}[\mathcal{S}])^{\frac{1}{2}}\mathcal{B}\mathcal{M}^{\dagger\frac{1}{2}}X\right\|^2 \\ \frac{1}{n}\left\|\mathcal{M}^{\dagger\frac{1}{2}}X\right\|^2 &\ge \frac{2\alpha}{n^2}\mathbb{E}\left[\|\mathcal{U}Xe\|^2\right] + \left\|(\mathbb{E}[\mathcal{S}])^{\frac{1}{2}}\mathcal{B}\mathcal{M}^{\dagger\frac{1}{2}}X\right\|^2 \end{aligned}$$

Theorem (simplified)

For `GJS` we have $\mathbb{E}\left[\Psi^k\right] \le (1-\alpha\sigma)^k\Psi^0$ for $\Psi^k \overset{\text{def}}{=} \left\|x^k - x^*\right\|^2 + \alpha\left\|\mathcal{B}\left(J^k - G(x^*)\right)\right\|^2$.

- Linear convergence under minimal assumptions
- Rate depends on smoothness patterns (matrices M_j), distributions of $\mathcal{S}$,$\mathcal{U}$ (controllable in practice) and quasi strong convexity σ
- **Full version:** exploits possible prior knowledge about $G(x^*)$, exploits structure of ψ, extends quasi strong convexity to strong growth.

Special cases

- `SAGA` [1]: recovers best known results
- `JacSketch` [2] More general + better rate
- `LSVRG`: Arbitrary sampling + prox
- `SEGA` [3]: Better rate under arbitrary sampling
- Extensions of algorithms from [4] – arbitrary sampling and conjectured `ISEAGA`.
- Many more:

Choice of random operators $\mathcal{S}$ and $\mathcal{U}$ defining Algorithm 1		Algorithm			
$\mathcal{S}\mathbf{X}$	$\mathcal{U}\mathbf{X}$	#	Name	Comment	Section
$\mathbf{X}e_je_j^\top$ w.p. $p_j = \frac{1}{n}$	$\mathbf{X}ne_je_j^\top$ w.p. $p_j = \frac{1}{n}$	2	`SAGA`	basic variant of `SAGA` [3]	G.1
$\mathbf{X}\sum_{j\in R} e_je_j^\top$ w.p. p_R	$\mathbf{X}\sum_{j\in R}\frac{1}{p_j}e_je_j^\top$ w.p. p_R	3	`SAGA`	`SAGA` with AS [26]	G.2
$e_ie_i^\top\mathbf{X}$ w.p. $p_i = \frac{1}{d}$	$de_ie_i^\top\mathbf{X}$ w.p. $p_i = \frac{1}{d}$	4	`SEGA`	basic variant of `SEGA` [9]	H.1
$\sum_{i\in L} e_ie_i^\top\mathbf{X}$ w.p. p_L	$\sum_{i\in L}\frac{1}{p_i}e_ie_i^\top\mathbf{X}$ w.p. p_L	5	`SEGA`	`SEGA` [9] with AS and prox	H.2
$= \begin{cases} 0 & \text{w.p. } 1-\rho \\ \mathbf{X} & \text{w.p. } \rho \end{cases}$	$\sum_{i\in L}\frac{1}{p_i}e_ie_i^\top\mathbf{X}$ w.p. p_L	6	`SVRCD`	**NEW**	H.3
0	$\mathbf{X}\sum_{j\in R}\frac{1}{p_j}e_je_j^\top$ w.p. p_R	7	`SGD-shift`	**NEW**	I
$= \begin{cases} 0 & \text{w.p. } 1-\rho \\ \mathbf{X} & \text{w.p. } \rho \end{cases}$	$\mathbf{X}\sum_{j\in R}\frac{1}{p_j}e_je_j^\top$ w.p. p_R	8	`LSVRG`	`LSVRG` [14] with AS and prox	J
$= \begin{cases} 0 & \text{w.p. } 1-\rho \\ \mathbf{X} & \text{w.p. } \rho \end{cases}$	$= \begin{cases} 0 & \text{w.p. } 1-\delta \\ \frac{1}{\delta}\mathbf{X} & \text{w.p. } \delta \end{cases}$	9	`B2`	**NEW**	K.1
$\mathbf{X}\sum_{j\in R} e_je_j^\top$ w.p. p_R	$= \begin{cases} 0 & \text{w.p. } 1-\delta \\ \frac{1}{\delta}\mathbf{X} & \text{w.p. } \delta \end{cases}$	10	`LSVRG-inv`	**NEW**	K.2
$\sum_{i\in L} e_ie_i^\top\mathbf{X}$ w.p. p_L	$= \begin{cases} 0 & \text{w.p. } 1-\delta \\ \frac{1}{\delta}\mathbf{X} & \text{w.p. } \delta \end{cases}$	11	`SVRCD-inv`	**NEW**	K.3
$\mathbf{X}\sum_{j\in R} e_je_j^\top$ w.p. p_R	$\sum_{i\in L}\frac{1}{p_i}e_ie_i^\top\mathbf{X}$ w.p. p_L	12	`RL`	**NEW**	L.1
$\sum_{i\in L} e_ie_i^\top\mathbf{X}$ w.p. p_L	$\mathbf{X}\sum_{j\in R}\frac{1}{p_j}e_je_j^\top$ w.p. p_R	13	`LR`	**NEW**	L.2
$\mathbf{I}_{L:}\mathbf{X}\mathbf{I}_{:R}$ w.p. p_Lp_R	$\mathbf{I}_{L:}\left(\left(p^{-1}\left(p^{-1}\right)^\top\right)\circ\mathbf{X}\right)\mathbf{I}_{:R}$ w.p. p_Lp_R	14	`SAEGA`	**NEW**	M.1
$= \begin{cases} 0 & \text{w.p. } 1-\rho \\ \mathbf{X} & \text{w.p. } \rho \end{cases}$	$\mathbf{I}_{L:}\left(\left(p^{-1}\left(p^{-1}\right)^\top\right)\circ\mathbf{X}\right)\mathbf{I}_{:R}$ w.p. p_Lp_R	15	`SVRCDG`	**NEW**	M.2
$\sum_{t=1}^T \mathbf{I}_{L_t:}\mathbf{X}_{:N_t}\mathbf{I}_{:R_t}$	$\sum_{t=1}^T\left((p^t)^{-1}(p^t)^{-1\top}\right)\circ\left(\mathbf{I}_{L_t:}\mathbf{X}_{:N_t}\mathbf{I}_{:R_t}\right)$	16	`ISAEGA`	**NEW** (reminiscent of [20])	M.3
$\sum_{t=1}^T \mathbf{I}_{L_t:}\mathbf{X}_{:N_t}$	$\sum_{t=1}^T\left((p^t)^{-1}e^\top\right)\circ\left(\mathbf{I}_{L_t:}\mathbf{X}_{:N_t}\right)$	17	`ISEGA`	`ISEGA` [20] with AS	M.3
$\mathbf{X}\mathbf{R}$	$\mathbf{X}\mathbf{R}\mathbb{E}[\mathbf{R}]^{-1}$	18	`JS`	`JacSketch` [8] with AS and prox	N

Arbitrary sampling

- Tight rate under any distribution of $\mathcal{S}$,$\mathcal{U}$
- Allows to exploit data structure from smoothness (matrices M_j) and design importance samplings
- New for many well established algorithms, bridged by our analysis

Specific algorithms

Algorithm 2 `SEGA` with arbitrary sampling

Require: Stepsize $\alpha > 0$, starting point $x^0 \in \mathbb{R}^d$, random sampling $L \subseteq \{1, 2, \dots, d\}$
  Set $h^0 = 0$
  for $k = 0, 1, 2, \dots$ **do**
    Sample random $L^k \subseteq \{1, 2, \dots, d\}$
    Set $h^{k+1} = h^k + \sum_{i\in L^k}(\nabla_i f(x^k) - h_i^k)e_i$
    $g^k = h^k + \sum_{i\in L^k}\frac{1}{p_i}(\nabla_i f(x^k) - h_i^k)e_i$
    $x^{k+1} = \operatorname{prox}_{\alpha\psi}(x^k - \alpha g^k)$
  end for

Algorithm 3 `ISAEGA` **[NEW METHOD]**

  Input: $x^0 \in \mathbb{R}^d$, # parallel units T, each owning set of indices N_t (for $1 \le t \le T$), distributions $\mathcal{D}_t$ over subsets of N_t, distributions $\mathcal{D}_t$ over subsets coordinates $[d]$, stepsize α
  $J^0 = 0$
  for $k = 0, 1, \dots$ **do**
    for $t = 1, \dots, T$ in parallel **do**
      Sample $R_t \sim \mathcal{D}_t$; $R_t \subseteq N_t$, $L_t \sim \mathcal{D}_t$; $L_t \subseteq [d]$
      Observe $\nabla_{L_t} f_j(x^k)$ for $j \in R_t$
      Set $J_{i,j}^{k+1} = \begin{cases} \nabla_i f_j(x^k) & \text{if } \ , j \in R_t, i \in L_t \\ J_{i,j}^k & \text{otherwise} \end{cases}$
      Send $J_{:N_t}^{k+1} - J_{:N_t}^k$ to master ▷ Sparse
    end for
    $g^k = \left(J^k + \sum_{t=1}^T\left(p^{t-1}p^{t-1\top}\right)\circ\left(\left(\sum_{i\in L_t}e_ie_i^\top\right)\left(J^{k+1} - J^k\right)_{:N_t}\left(\sum_{j\in R_t}e_je_j^\top\right)\right)\right)e$
    $x^{k+1} = \operatorname{prox}_{\alpha\psi}(x^k - \alpha g^k)$
  end for

References

[1] Aaron Defazio, Francis Bach, and Simon Lacoste-Julien.
SAGA: A fast incremental gradient method with support for non-strongly convex composite objectives.
In *Advances in Neural Information Processing Systems*, pages 1646–1654, 2014.

[2] Robert M Gower, Peter Richtárik, and Francis Bach.
Stochastic quasi-gradient methods: Variance reduction via Jacobian sketching.
arXiv preprint arXiv:1805.02632, 2018.

[3] Filip Hanzely, Konstantin Mishchenko, and Peter Richtárik.
SEGA: Variance reduction via gradient sketching.
In *Advances in Neural Information Processing Systems*, pages 2082–2093, 2018.

[4] Konstantin Mishchenko, Filip Hanzely, and Peter Richtárik.
99% of distributed optimization is a waste of time: The issue and how to fix it.
arXiv preprint arXiv:1901.09437, 2019.