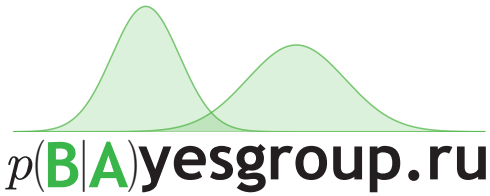
Optimization Methods for Big Sums of Functions

Anton Rodomanov

Higher School of Economics



Bayesian methods research group
(http://bayesgroup.ru)

5 June 2016

Skoltech Deep Machine Intelligence Workshop, Moscow, Russia

Introduction

Consider the problem

$$\text{Find } f^* = \min_{x \in \mathbb{R}^d} f(x) \quad \text{with } f(x) := \frac{1}{n} \sum_{i=1}^{n} f_i(x),$$

Example (Empirical risk minimization):

- We are given observations a_i (and possibly their labels β_i).
- Goal: find optimal parameters x^* of a parametric model.
- Linear regression ($a_i \in \mathbb{R}^d,\ \beta_i \in \mathbb{R}$):

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} \left\| a_i^\top x - \beta_i \right\|^2$$

- Logistic regression ($a_i \in \mathbb{R}^d,\ \beta_i \in \{-1, 1\}$):

$$f(x) = \frac{1}{n} \sum_{i=1}^{n} \ln(1 + \exp(-\beta_i a_i^\top x))$$

- Neural networks, SVMs, CRFs etc.

Preliminaries

Problem: $f^* = \min_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$

Goal: Given $\epsilon > 0$, find $\bar{x}$ such that $f(\bar{x}) - f^* \leq \epsilon$.

Assumptions:

- Each function f_i is *L-smooth*:
$$\|\nabla f_i(x) - \nabla f_i(y)\| \leq L \|x - y\|, \qquad \forall x, y \in \mathbb{R}^d.$$
- Function f is *μ-strongly convex*:
$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle + \frac{\mu}{2} \|y - x\|^2, \qquad \forall x, y \in \mathbb{R}^d.$$

Strong convexity of f implies existence of a unique $x^* : f(x^*) = f^*$.
We consider iterative methods which produce $\{x^k\}_{k \geq 0} : x^k \to x^*$.

Gradient descent and big sums of functions

Problem: $f^* = \min_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$

Gradient descent:

$$x^{k+1} = x^k - \eta \nabla f(x^k)$$

$$\nabla f(x^k) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x^k)$$

Here $\eta \in \mathbb{R}_{++}$ is a step length.

Note:

- Computation of $\nabla f(x^k)$ requires $O(nd)$ operations.
- When n is very large, this may take a lot of time. Example: $n = 10^8$, $d = 1000 \Rightarrow$ evaluating $\nabla f(x^k)$ takes ≥ 2 minutes.
- We need methods with cheaper iterations.

Stochastic gradient descent [Robbins & Monro, 1951]

Problem: $f^* = \min_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$

Stochastic Gradient Descent (SGD):

Choose $i_k \in \{1, \ldots, n\}$ uniformly at random

$$x^{k+1} = x^k - \eta_k \nabla f_{i_k}(x^k).$$

Here $\{\eta_k\}_{k \geq 0} \subseteq \mathbb{R}_{++}$ is a sequence of step lengths converging to 0.

Motivation: $\mathbb{E}_{i_k}[\nabla f_{i_k}(x^k)] = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(x^k) = \nabla f(x^k)$, i.e., on average, SGD makes a step in the right direction.

Note:

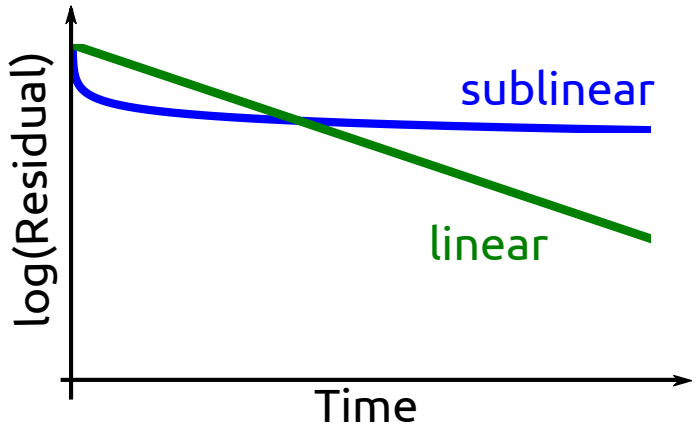
- Now we only need to compute one gradient instead of n.
- Iteration complexity: $O(d)$. Independent of n!
- No reliable stopping criterion (cannot compute $\|\nabla f(x_k)\|$).

Gradient descent vs SGD: Which one is better?

Problem: $f^* = \min_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$

Iteration cost:

- Gradient descent: $O(nd)$.
- SGD: $O(d)$.



Convergence rate:

- Gradient descent: *linear*, $O\left(nd\frac{L}{\mu}\ln\frac{1}{\epsilon}\right)$ flops for ϵ-solution.
- SGD: *sublinear*, $O(\frac{d}{\mu\epsilon})$ flops for ϵ-solution.

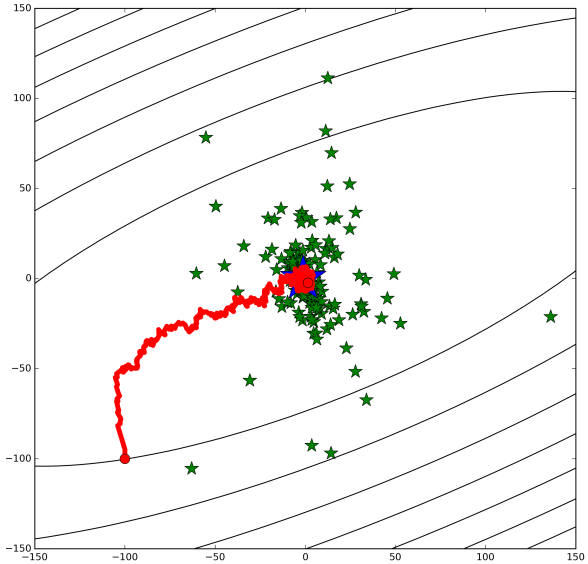
Discussion:

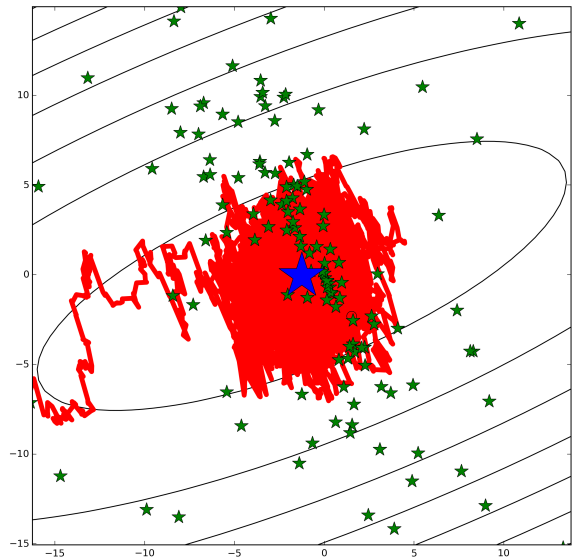
- Complexity of SGD does not depend on n.
- SGD is good for large ϵ and terrible for small ϵ.

Slow convergence of SGD: Why?

Problem: $f^* = \min\limits_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum\limits_{i=1}^{n} f_i(x).$

Example (Least squares): $f_i(x) := (a_i^\top x - b_i)^2$





Main reason for slow convergence of SGD is the variance

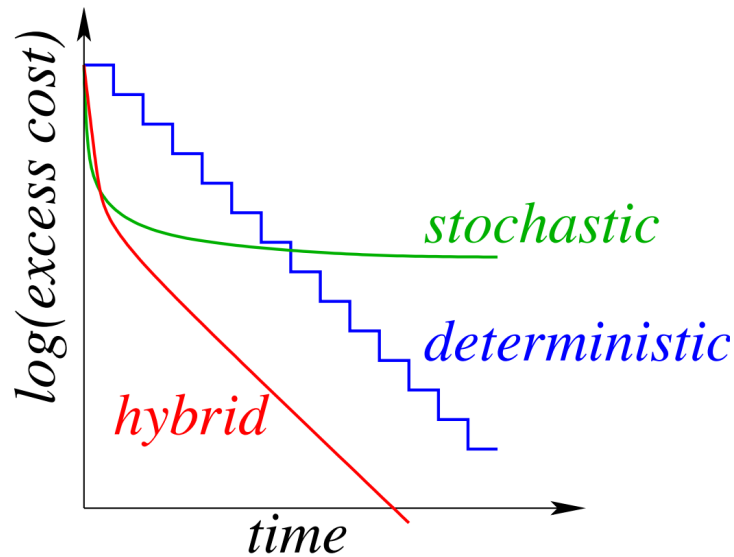
$$\sigma_k^2 := \mathbb{E}_i \left[\left\| \nabla f_i(x^k) - \nabla f(x^k) \right\|^2 \right].$$

Note that even if $x^k \to x^*$ we have $\sigma_k \to \sigma > 0$.

Towards a hybrid method

Gradient descent: $O(nd)$ iteration cost, linear convergence.
SGD: $O(d)$ iteration cost, sublinear convergence.

Goal: $O(d)$ iteration cost, linear convergence.



Credit: *Nicolas Le Roux et al.*

Methods: SAG [Le Roux et al., 2012], SVRG [Johnson & Zhang, 2013], SAGA [Defazio et al., 2014a], MISO [Mairal, 2015] etc.
We only consider SVRG as the most practical one for a general f_i.

Main idea: variance reduction, $\mathbb{E}_i[\|g_i^k - \nabla f(x^k)\|^2] \to 0$.

Stochastic Variance Reduced Gradient [Xiao & Zhang, 2014]

Problem: $f^* = \min_{x \in \mathbb{R}^d} f(x), \quad f(x) = \frac{1}{n} \sum_{i=1}^{n} f_i(x).$

Require: $\tilde{x}^0$: initial point; m: update frequency; η: step length.

for $s = 0, 1, \ldots$ **do**

$\tilde{g}^s := \nabla f(\tilde{x}^s) = \frac{1}{n} \sum_{i=1}^{n} \nabla f_i(\tilde{x}^s)$

$x^0 := \tilde{x}^s$

for $k = 0, \ldots, m-1$ **do**

Choose $i_k \in \{1, \ldots, n\}$ uniformly at random

$x^{k+1} := x^k - \eta(\nabla f_{i_k}(x^k) - \nabla f_{i_k}(\tilde{x}^s) + \tilde{g}^s)$

end for

$\tilde{x}^{s+1} := \frac{1}{m} \sum_{k=1}^{m} x^k \qquad (\text{or } \tilde{x}^{s+1} := x^m)$

end for

Parameters: usually $m = O(n)$, $\eta = O(\frac{1}{L})$; e.g. $m = 2n$, $\eta = \frac{1}{10L}$.

Note:

- Works with a constant step length.
- Reliable stopping criterion: $\|\tilde{g}^s\|^2 \leq \tilde{\epsilon}$.

Variance reduction in SVRG

Denote $g_i := \nabla f_i(x) - \nabla f_i(\tilde{x}) + \nabla f(\tilde{x})$.
Then g_i is an unbiased estimate of $\nabla f(x)$:

$$\mathbb{E}_i[\nabla f_i(x) - \nabla f_i(\tilde{x}) + \nabla f(\tilde{x})] = \nabla f(x) - \nabla f(\tilde{x}) + \nabla f(\tilde{x}) = \nabla f(x).$$

Variance:

$$\begin{aligned}
\sigma^2 &:= \mathbb{E}_i\left[\|g_i - \nabla f(x)\|^2\right] \\
&= \mathbb{E}_i\left[\|(\nabla f_i(x) - \nabla f_i(\tilde{x})) - (\nabla f(x) - \nabla f(\tilde{x}))\|^2\right] \\
&\qquad (\|a+b\|^2 \le 2\|a\|^2 + 2\|b\|^2) \\
&\le 2\mathbb{E}_i\left[\|\nabla f_i(x) - \nabla f_i(\tilde{x})\|^2\right] + 2\|\nabla f(x) - \nabla f(\tilde{x})\|^2 \\
&\le 2L^2\|x - \tilde{x}\|^2 + 2L^2\|x - \tilde{x}\|^2 \\
&= 4L^2\|x - \tilde{x}\|^2.
\end{aligned}$$

Note: when $x \to x^*$ and $\tilde{x} \to x^*$, then $\sigma \to 0$.
In plain SGD we had $g_i = \nabla f_i(x)$ and so $\sigma \not\to 0$ when $x \to x^*$.

SVRG: Convergence analysis [Xiao & Zhang, 2014]

Theorem

Let $\eta < \frac{1}{4L}$ and m is sufficiently large so that

$$\rho := \frac{1}{\mu\eta(1-4L\eta)m} + \frac{4L\eta(m+1)}{(1-4L\eta)m} < 1.$$

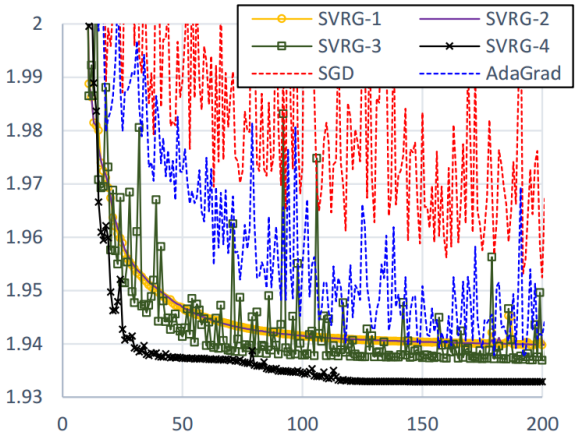
Then SVRG converges at a linear rate:

$$\mathbb{E}[f(\tilde{x}^s)] - f^* \leq \rho^s [f(\tilde{x}^0) - f^*].$$

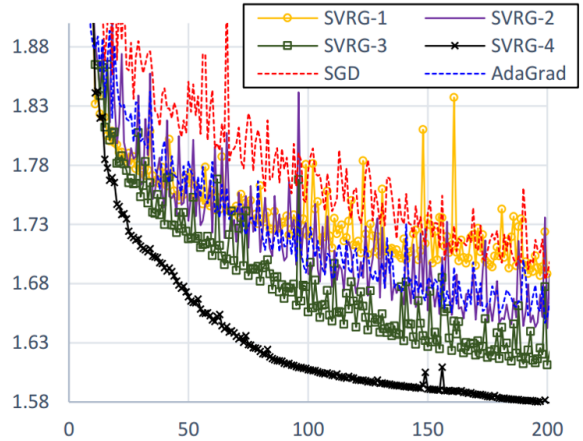
Discussion:

- Let us choose $\eta = \frac{1}{10L}$ and assume $m \gg 1$. Then $4L\eta = \frac{2}{5}$ and
$$\rho \approx \frac{50\frac{L}{\mu}}{3m} + \frac{2}{3}$$
- To ensure $\rho < 1$, let us choose $m = 100\frac{L}{\mu}$. Then $\rho \approx \frac{5}{6}$.
- To reach ϵ, we need to perform $s = O(\ln \frac{1}{\epsilon})$ epochs.
- Complexity of each epoch: $O((n+m)d) = O((n+\frac{L}{\mu})d)$.
- Thus total complexity is $O\left((n+\frac{L}{\mu})d \ln \frac{1}{\epsilon}\right)$.
- Recall that for gradient descent we had $O\left((n\frac{L}{\mu})d \ln \frac{1}{\epsilon}\right)$.

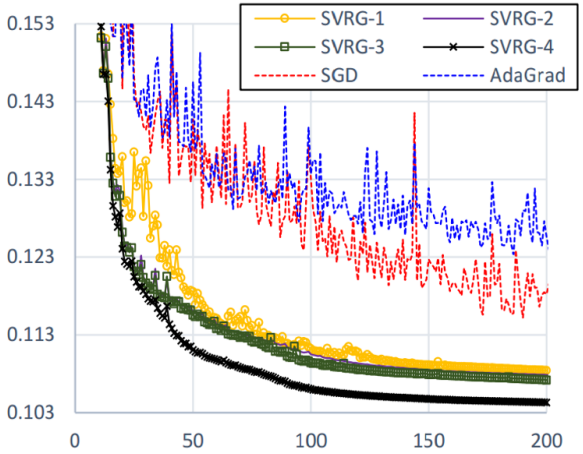
Practical performance [Allen-Zhu & Hazan, 2016]



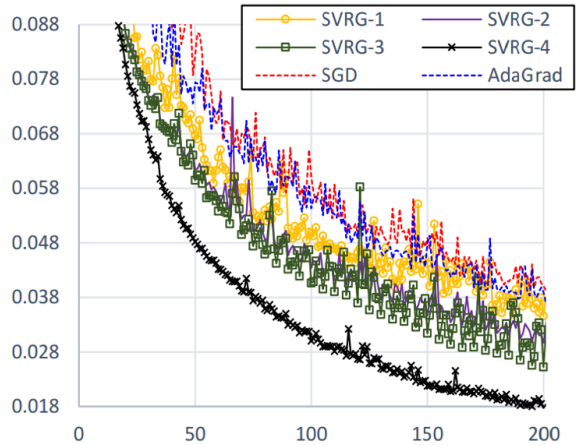
(b) cifar-10, $\lambda = 10^{-3}$



(a) cifar-10, $\lambda = 0$



(d) mnist, $\lambda = 10^{-4}$



(c) mnist, $\lambda = 0$

Figure: Training Error Comparison on neural nets. Y axis: training objective value; X axis: number of passes over dataset.

Conclusion

- SGD is a general method which is suitable for any stochastic optimization problem.
- However, SGD has a sublinear rate of convergence. The main reason for that is the large variance in estimating the gradient which does not decrease with time.
- For the special case of finite sums of functions it is possible to design SGD-like methods which reduce the variance when they progress. This allows them to achieve a linear rate of convergence.
- This variance reduction has an effect only after multiple passes through the data.
- If one can perform only a couple passes through the data, then SGD is an optimal method. If several passes through the data are allowed, variance reducing methods (e.g. SVRG) work much better.

Thank you!

References

- Original paper on SVRG:

  R. Johnson & T. Zhang. Accelerating Stochastic Gradient Descent using Predictive Variance Reduction, NIPS 2013.

- SVRG for composite functions:

  L. Xiao & T. Zhang. A Proximal Stochastic Gradient Method with Progressive Variance Reduction. SIAM Journal on Optimization, 2014.

- Practical improvements for SVRG:

  R. Babanezhad et al.. Stop Wasting My Gradients: Practical SVRG, NIPS 2015.

- Theory of SVRG for non-strongly convex and non-convex functions:
  - J. Reddi et al.. Stochastic Variance Reduction for Nonconvex Optimization, ICML 2016.
  - Z. Allen-Zhu & Y. Yuan. Improved SVRG for Non-Strongly-Convex or Sum-of-Non-Convex Objectives, ICML 2016.
  - Z. Allen-Zhu & E. Hazan. Variance Reduction for Faster Non-Convex Optimization, ICML 2016.