IFT 6085 - Lecture 15
Variance Reduction Methods for Stochastic Optimization

Scribe(s): Aristide Baratin, Gabriel Huang **Instructor:** Ioannis Mitliagkas

1 Summary

In the previous lecture we, discussed aspects of the training dynamics of GANs. We have seen in particular that, in this setting, the convergence rate is dictated by the largest (absolute) eigenvalue, and that maintaining small eigenvalues requires using a small learning rate. One way to allow larger learning rate is to add a gradient penalty.

In this lecture we introduce variance reduced methods, which give yet another way to stabilize training with larger learning rates. These methods attempt at combining the best properties of both stochastic gradient descent (SGD) and (full batch) gradient descent (GD).

2 SGD versus GD

Stochastic gradient methods are particularly well-suited for optimization problems of the form

$$\min_x f(x) = \frac{1}{n}\sum_i f_i(x)$$

in situations where the number n of data points is very large. While the iterations of (full batch) gradient descent take the form

$$x_{k+1} = x_k - \gamma \nabla_x f(x),$$

the iterations SGD use (cheap) unbiased estimates of the full gradient $\nabla_x f(x)$.

Definition 1 (SGD)**.** *SGD iterations typically take the form*

$$x_{k+1} = x_k - \gamma \nabla_x f_{s_k}(x)$$

where $\nabla_x f_{s_k}(x)$ is an unbiased estimate of the full gradient $\nabla_x f(x)$ obtained by sampling uniformly a random index $s_k \in \{1, .., n\}$; and γ is the step size.

The main advantage of SGD over GD is a gain of iteration cost by a factor $1/n$. However the effect of noisy gradient estimates requires decreasing step-sizes, which slows down the convergence.

Theorem 2 (Convergence rates)**.** *Under the assumption that the objective is strongly convex,*

- *GD converges in $O(\rho^T)$ for some $\rho \in (0, 1)$ (linear convergence), with an iteration cost in $O(n)$*
- *SGD converges in $O(1/T)$ (sublinear convergence), with an iteration cost in $O(1)$*

where T denotes the number of iterations.

Several methods have been recently designed [7, 3, 5] to reduce the variance of the gradient estimates and accelerate convergence of SGD. The main idea is to find better search directions by storing and re-using gradient estimates of previous iterations. These methods speed up convergence from sublinear to linear for strongly convex problems, i.e make SGD reach the convergence rate of GD (while maintaining cheap iteration cost).

3 Background

We first recall some basic notations and definitions.

Definition 3 (Strong convexity)**.** . *The objective function $f\colon \mathbb{R}^d \to \mathbb{R}$ is λ-strongly convex if*

$$\langle \nabla f(x) - \nabla f(y), x - y \rangle \geq \lambda ||x - y||^2$$

for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$.

Definition 4 (Smoothness)**.** . *The objective function $f\colon \mathbb{R}^d \to \mathbb{R}$ is β-smooth if*

$$||\nabla f(x) - \nabla f(y)|| \leq \beta ||x - y||$$

for all $(x, y) \in \mathbb{R}^d \times \mathbb{R}^d$.

Definition 5 (Bounding the second moment)**.** . *We say that the gradient estimate $\nabla f_s(x)$ has bounded second moment if there exists $M > 0$ such that*

$$\mathbb{E}_s ||\nabla f_s(x)||^2 \leq M^2$$

for all $x \in \mathbb{R}^d$.

4 Proof of Theorem 2

In this section we give proofs of the convergence rates of GD and SGD stated in Theorem 2. We assume that the objective function is **λ-strongly convex** and **β-smooth**.

4.1 Convergence rate of GD

In what follow we denote by D_k the (square) **distance to the optimum** at iteration k:

$$D_k = ||x_k - x^*||^2$$

To establish the linear convergence rate, the strategy is to find $\rho \in (0, 1)$ such that $D_{k+1} \leq \rho D_k$ for all $k \in \mathbb{N}$. Iterating this inequality T times would then give $D_T \leq \rho^T D_0$, hence a convergence rate in $O(\rho^T)$.

We have that

$$\begin{aligned} D_{k+1} &= ||x_{k+1} - x^*||^2 \\ &= ||x_k - \gamma \nabla f(x_k) - x^*||^2 \\ &= ||x_k - x^*||^2 - 2\gamma \langle x_k - x^*, \nabla f(x_k) \rangle + \gamma^2 ||\nabla f(x_k)||^2 \end{aligned}$$

where the second line used the expression of the GD iteration. Now, using λ-strong convexity to bound the second termm, we obtain:

$$\begin{aligned} D_{k+1} &\leq (1 - 2\gamma\lambda) D_k + \gamma^2 ||\nabla f(x_k)||^2 \\ &= (1 - 2\gamma\lambda) D_k + \gamma^2 ||\nabla f(x_k) - \nabla f(x^*)||^2 \end{aligned}$$

where the second line simply introduced the term $\nabla f(x^*)$, which is zero since the gradient vanishes at the optimum. Finally, β-smoothness allows us to bound the second term of the right hand side by $\gamma^2 \beta^2 D_k$, so we conclude

$$D_{k+1} \leq \rho\, D_k \qquad \text{with } \rho = 1 - 2\gamma\lambda + \gamma^2\beta^2$$

This guarantees a linear convergence rate as long as the step size satisfies $\gamma \leq 2\lambda/\beta^2$.

4.2 The case of SGD

How to adapt the proof for SGD? We can try to reproduce the above inequalities *in expectation* over the choice of index samples s_k. Note that we could also instead prove that convergence bounds hold *with high-probability*.

The first three equalities of the previous proof, as well λ-strong convexity inequality, are preserved through expectations, so that

$$\mathbf{E}[D_{k+1}] \leq (1 - 2\gamma\lambda)\mathbf{E}[D_k] + \gamma^2 \mathbf{E}||\nabla f_{s_k}(x_k)||^2$$

However we **cannot** go further and directly write:

$$\mathbf{E}[D_{k+1}] \leq (1 - 2\gamma\lambda)\mathbf{E}[D_k] + \gamma^2 \mathbf{E}||\nabla f_{s_k}(x_k) - \nabla f_{s_k}(x^*)||^2$$

because the stochastic gradient $\nabla f_{s_k}(x^*)$ is generally **not** zero at the optimum.

Instead, we can add and subtract the term:

$$\mathbf{E}[D_{k+1}] \leq (1 - 2\gamma\lambda)\mathbf{E}[D_k] + \gamma^2 \mathbf{E}||\nabla f_{s_k}(x_k) - \nabla f_{s_k}(x^*) + \nabla f_{s_k}(x^*)||^2$$

Now we use the identity $||a + b||^2 \leq 2||a||^2 + 2||b||^2$ and get:

$$\mathbf{E}[D_{k+1}] \leq (1 - 2\gamma\lambda)\mathbf{E}[D_k] + 2\gamma^2 \mathbf{E}||\nabla f_{s_k}(x_k) - \nabla f_{s_k}(x^*)||^2 + 2\mathbf{E}||\nabla f_{s_k}(x^*)||^2$$

Finally by β-smoothness we obtain:

$$\mathbf{E}[D_{k+1}] \leq (1 - 2\gamma\lambda + 2\beta^2\gamma^2)\mathbf{E}[D_k] + 2\gamma^2 \mathbf{E}||\nabla f_{s_k}(x^*) - \nabla f(x^*)||^2 \qquad (*)$$

The last term above is the **noise ball** (due to the gradient at optimum having nonzero variance). At that point, to continue and reach convergence, we need to reduce learning rate – which considerably slows down the convergence.

One solution to this problem is to rely on variance reduction methods.

5 Variance reduction methods

The goal of variance reduction methods is define unbiased updates:

$$x_{k+1} = x_k - \gamma v_k(x_k)$$

with less variance, i.e., $Var(v_k)$ is small.

The idea is to introduce an extra-term in the SGD updates, which has zero mean but will lower the variance of the update:

$$v_k(x_k) = \nabla f_{s_k}(x_k) - \underbrace{\nabla f_{s_k}(y) + \nabla f(y)}_{\text{new term}}$$

where y is a generic point.

Let us compute the variance of v_k:

$$\begin{aligned}
\mathbf{E}_{s_k}||v_k||^2 &= \mathbf{E}||\nabla f_{s_k}(x) - \nabla f_{s_k}(y) + \nabla f(y)||^2 \\
&= \mathbf{E}||\nabla f_{s_k}(x) - \nabla f_{s_k}(y) + \nabla f(y) + \nabla f_{s_k}(x^*) - \nabla f_{s_k}(x^*)||^2 \\
&\leq 2\mathbf{E}||\nabla f_{s_k}(x) - \nabla f_{s_k}(x^*)||^2 + 2\mathbf{E}||\underbrace{\nabla f_{s_k}(y) - \nabla f_{s_k}(x^*)}_{\mathbf{E}[[\ldots]] = \nabla f(y)} - \nabla f(y)||^2
\end{aligned}$$

Using the inequality $\mathbf{E}||X - \mathbf{E}X||^2 \leq \mathbf{E}||X||^2$, we obtain:

$$\mathbf{E}_s||v_k||^2 \leq 2\mathbf{E}||\nabla f_{s_k}(x) - \nabla f_{s_k}(x^*)||^2 + 2\mathbf{E}||\nabla f_s(y) - \nabla f_s(x^*)||^2$$

We can now use β-smoothness to bound the two terms of the sum, so that

$$\mathbf{E}_{s_k}||v_k||^2 \leq 2\beta^2 \mathbf{E}[D_k] + 2\beta^2 \mathbf{E}||y - x^*||^2$$

Example Let $y = x_{s_1}$. In this case we get

$$\mathbf{E}_s||v_k||^2 \leq 2\beta^2\mathbf{E}[D_k] + 2\beta^2\mathbf{E}D_1$$

Redoing the analysis of Section 4.2 by replacing the gradients with the updates v_k, the last equation $(*)$ becomes:

$$\begin{aligned}\mathbf{E}[D_{k+1}] &\leq (1 - 2\gamma\lambda + 2\beta^2\gamma^2)\mathbf{E}[D_k] + 2\gamma^2\mathbf{E}||v_k||^2 \\ &\leq (1 - 2\gamma\lambda + 2\beta^2\gamma^2)\mathbf{E}D_k + 2\gamma^2 2\beta^2\mathbf{E}D_k + 2\gamma^2 2\beta^2\mathbf{E}D_1 \\ &\leq (1 - 2\gamma\lambda + 6\beta^2\gamma^2)\mathbf{E}D_k + 4\gamma^2\beta^2\mathbf{E}D_1\end{aligned}$$

Let $\rho = 1 - 2\gamma\lambda + 6\beta^2\gamma^2$. We have that $\rho \in (0, 1)$ whenever $\gamma \leq \lambda/\beta^2$. Applying the previous inequality T times, we obtain after T steps,

$$\mathbf{E}D_{T+1} \leq (\rho^T + T4\gamma^2\beta^2)\mathbf{E}D_1$$

This allows to answer guarantees questions such as: the number of updates needed and how to adjust the learning rate so that, for instance

$$\mathbf{E}D_{T+1} \leq 0.5\mathbf{E}D_1$$

We can take $\gamma = O(\lambda/\beta^2)$, $T = O(\beta^2/\lambda^2) = O(\kappa^2)$ where κ is the condition number. Then to have

$$\mathbf{E}D_{ET} \leq 0.5^E\mathbf{E}D_1$$

we can take ???

The idea of Stochastic Variance Reduced Gradient (SVRG) [5] is thus to compute the full gradient $\nabla f(\tilde{y})$ every T iterations (i.e at $y = x_k$ for $k = 1, T + 1, 2T + 1, \cdot$ and use it (in an inner loop) to correct the usual stochastic gradient estimate with a control variate $\nabla f_{s_k}(y) - \nabla f(y)$. The algorithm is as follows.

Algorithm 1 SVRG

Input x_1
$y \leftarrow x_1$, $k \leftarrow 1$
For epochs $e = 1 \cdots E$:
  Compute full gradient $g \leftarrow \nabla f(y)$
  For $k = 1 \cdots T$:
    sample $s_k \sim$ Uniform$\{1, \cdots n\}$
    Compute SVRG updates: $x_k \leftarrow x_k - \gamma(\nabla f_{s_k}(x_k) - \nabla f_{s_k}(y) + g)$
  $y \leftarrow x_k$

Remark: On non-convex functions, the estimate might be outdated.
There were several extensions to SVRG; [4] proposes to compute the gradient of growing-batches in the early iteration instead of computing the full batch gradient, [6] extends the methods to mini-batch, [2] improves SVRG by using increasing epoch size, finally closest to our work [1] proposed to use adaptive learning rate with SVRG for non-convex optimization.

6 Comparisons

Table 1 below compares full batch, stochastic and variance-reduced methods in terms of cost and speed. We assume the unit cost is 1 dollar per gradient evaluation on single example. We compare the number of iterations required to reach ϵ-suboptimality.

We see that SVRG is better than GD when $n + \kappa \leq n\kappa$, that is when the problem is well conditioned (κ is high).When aiming for a tiny error ϵ, SVRG beats SGD, as $\frac{1}{\epsilon}$ becomes large.

Side remark: An important question is then: what is a good ϵ for machine learning?

Algorithm	Iterations	Cost
SGD	$T = O(\frac{\kappa}{\epsilon})$	$O(\frac{\kappa}{\epsilon})$
GD	$T = O(\kappa \log \frac{1}{\epsilon})$	$O(n\kappa \log \frac{1}{\epsilon})$
SVRG	$T = O(\log \frac{1}{\epsilon})$	$O((n + \kappa^2)\kappa \log \frac{1}{\epsilon})$ or $O((n + \kappa)\kappa \log \frac{1}{\epsilon})$ or better $O((n + \kappa)\kappa \log \frac{1}{\epsilon})$

Table 1: Comparison of full batch, stochastic, and variance reduced methods

References

[1] Z. Allen-Zhu and E. Hazan. Variance reduction for faster non-convex optimization. *arXiv preprint arXiv:1603.05643*, 2016.

[2] Z. Allen-Zhu and Y. Yuan. Improved svrg for non-strongly-convex or sum-of-non-convex objectives. Technical report, Technical report, arXiv preprint, 2016.

[3] A. Defazio, F. Bach, and L.-J. Simon. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *Advances In Neural Information Processing Systems*, 2014.

[4] R. Harikandeh, M. O. Ahmed, A. Virani, M. Schmidt, J. Konecny, and S. Sallinen. Stop wasting my gradients: Practical svrg. In *Advances in Neural Information Processing Systems*, pages 2251–2259, 2015.

[5] R. Johnson and T. Zhang. Accelerating stochastic gradient descent using predictive variance reduction. *Advances in Neural Information Processing Systems*, pages 315–323, 2013.

[6] J. Konecny, J. Liu, P. Richtarik, and M. Takac. Mini-batch semi-stochastic gradient descent in the proximal setting. *IEEE Journal of Selected Topics in Signal Processing*, 10(2):242–255, 2016.

[7] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *Mathematical Programming*, 162(1):83–112, 2017.