Beyond the Regret Minimization Barrier: Optimal Algorithms for Stochastic Strongly-Convex Optimization

Elad Hazan, Satyen Kale
Presenter: Zhe Li

December 3, 2015

Three main contributions

- The convergence rate for stochastic strongly-convex optimization: $O(\frac{\log(T)}{T}) \rightarrow O(\frac{1}{T})$
- The regret in the online stochastic strongly-convex optimization:$\Omega(\log(T))$
- The convergence rate for stochastic strongly-convex optimization from online-to-batch conversion is suboptimal

Main assumptions

- A convex and differentiable function $\mathcal{R}(\cdot)$ and corresponding Bregman divergence

$$B_{\mathcal{R}}(y, x) := \mathcal{R}(y) - \mathcal{R}(x) - \nabla\mathcal{R}(x) \cdot (y - x)$$

  $\mathcal{R}$ is strongly-convex w.r.t the norm $||\cdot||$, then $B_{\mathcal{R}}(y, x) \geq \frac{1}{2}||y - x||^2$

- F is λ-strongly convex w.r.t $B_{\mathcal{R}}$, i.e.

$$F(\alpha x + (1 - \alpha y)) \leq \alpha F(x) + (1 - \alpha)F(y) - \lambda\alpha(1 - \alpha)B_{\mathcal{R}}(y, x)$$

  which implies $F(x) - F(x^*) \geq \lambda B_{\mathcal{R}}(x^*, x)$

Main assumptions

- G-bound

$$E[||\hat{g}||_*^2] \leq G^2$$

and strongly G-bound

$$E[exp(\frac{||\hat{g}||_*^2}{G^2})] \leq exp(1)$$

- The Fenchel conjugate of $\mathcal{R}(\cdot)$ is the function $\mathcal{R}^*(\cdot)$

$$\mathcal{R}(\cdot)^* := \sup_x w \cdot x - \mathcal{R}(x)$$

By the properties of Fenchel conjugacy, $\nabla\mathcal{R}^* = \nabla\mathcal{R}^{-1}$

Main theorems

Theorem 1

Assume that F is λ-strongly convex and the gradient oracle is $G-$bounded. Then exists a deterministic algorithm that after at most T gradient updates returns a vector $\bar{x}$ such that for any $x^* \in \mathcal{K}$ we have

$$E[F(\bar{x})] - F(x^*) \leq O(\frac{G^2}{\lambda T}) \tag{1}$$

Theorem 2

For any online decision-making algorithm $\mathcal{A}$, there is a distribution over $\lambda-$strongly-convex cost functions with norms of gradients bounded by G such that

$$E[\text{Regret}(\mathcal{A})] = \Omega(\frac{G^2 \log(T)}{\lambda}) \tag{2}$$

Formally, as theorems state

Theorem 3

Assume that F is λ-strongly convex and the gradient oracle is strongly $G-$bounded. Then for any $\delta > 0$, there exists an algorithm that after at most T gradient updates returns a vector $\bar{x}$ such that with probability at least $1-\delta$, for any $x^* \in \mathcal{K}$ we have

$$F(\bar{x}) - F(x^*) \le O\left(\frac{G^2(\log(\frac{1}{\delta})) + \log\log(T)}{\lambda T}\right) \tag{3}$$

Epoch-GD

Input: parameters η_1, T_1, and total time T

Initialize $x_1^1 \in \mathcal{K}$ arbitrarily, and set $k = 1$

while $\sum_{i=1}^{k} T_i \leq T$

- for $t = 1$ to T_k do
  - Query the gradient oracle at x_t^k to obtain $\hat{g}_t$
  - Update

$$\mathbf{y}_{t+1}^k = \nabla R^*(\nabla R(x_t^k) - \eta_k \hat{g}_t) \quad \text{update step}$$

$$\mathbf{x}_{t+1}^k = \underset{x \in \mathcal{K}}{argmin}\{B_R(x, y_{t+1}^k)\} \quad \text{project step}$$

- End for
- Set $x_1^{k+1} = \frac{1}{T_k} \sum_{t=1}^{T_k} x_t^k$ and $T_{k+1} \leftarrow 2T_k; \eta_{k+1} \leftarrow \eta_k/2$
- Set $k \leftarrow k + 1$

End while and return x_1^k

Main theorem

Theorem 4

Set the parameters $T_1 = 4$ and $\eta_1 = \frac{1}{\lambda}$ in the EPOCH-GD algorithm. The final point x_1^k returned by algorithm has the property that

$$E[F(x_1^k)] - F(x^*) \leq \frac{16G^2}{\lambda T} \tag{4}$$

The total number of gradient updates is at most T.

Lemma used to proved theorem 4

Lemma

Starting from arbitrary point $x_1 \in \mathcal{K}$, apply T iterations of the update

$$y_{t+1} = \nabla R^*(\nabla R(x_t) - \eta \hat{g}_t)$$
$$x_{t+1} = \underset{x \in \mathcal{K}}{argmin} B_R(x, y_{t+1})$$

Then for any point $x^* \in \mathcal{K}$, we have

$$\sum_{t=1}^{T} \hat{g}_t \cdot (x_t - x^*) \leq \frac{\eta}{2} \sum_{t=1}^{T} ||\hat{g}_t||_*^2 + \frac{B_R(x^*, x_1)}{\eta} \quad (5)$$

Lemma used to proved theorem 4

Lemma

Starting from arbitrary point $x_1 \in \mathcal{K}$, apply T iterations of the update

$$y_{t+1} = \nabla R^*(\nabla R(x_t) - \eta \hat{g}_t)$$
$$x_{t+1} = \underset{x \in \mathcal{K}}{argmin} B_R(x, y_{t+1})$$

Where $\hat{g}_t$ is an unbiased estimator for a subgradient g_t of F at x_t satisfying assumption, then for any point $x^* \in \mathcal{K}$, we have

$$\frac{1}{T} E[\sum_{t=1}^{T} F(x_t) - F(x^*) \leq \frac{\eta}{2} G^2 + \frac{B_R(x^*, x_1)}{\eta T} \tag{6}$$

By convexity of F, we have the same bound for $E[F(\bar{x})] - F(x^*)$, where $\bar{x} = \frac{1}{T} \sum_{t=1}^{T} x_t$.

Lemma used to proved theorem 4

Lemma

Define $V_k = \frac{G^2}{2^{k-2}\lambda}$, then for any k, we have $E[\triangle_k] \leq V_k$

Lemma

For all $x \in \mathcal{K}$ and x^* the minimizer of F, we have $F(x) - F(x^*) \leq \frac{2G^2}{\lambda}$.

Algorithm 1 Randomized EPOCH-GD

1. **Input**: parameters η_1, T_1 and total time T
2. **Initialize**: $x_1 \in \mathcal{K}$ arbitrary, and set $k = 1, B_1 = 1, B_2 \in 1, 2, \cdots, T_1$ uniformly at random.
3. **for** $t = 1, 2, \cdots$ **do**
4.   **if** $t == B_{k+1}$ **then**
5. $k \leftarrow k + 1, T_k \leftarrow 2T_{k-1}$
6. $\eta_k \leftarrow \eta_{k+1}/2,\ B_{k+1} \in \{B_k, B_k + 1, \cdots, B_k + T_k - 1\}$
7.   **if** $B_{k+1} > T$ **then**
8.   Break **for** loop
9.   **end if**
10.   **end if**
11.   Query the gradient oracle at x_t to obtain $\hat{g}_t$
12.   Update $\mathbf{y}_{t+1}^k = \nabla R^*(\nabla R(x_t^k) - \eta_k \hat{g}_t)$
13.   Project $x_{t+1}^k = \underset{x \in \mathcal{K}}{argmin}\{B_R(x, y_{t+1}^k)\}$
14. **end for**
15. **Return:** x_t

Main theorem

Theorem 5

Set the parameters $T_1 = 4$ and $\eta_1 = \frac{1}{\lambda}$ in the RANDOM-STEP-GD algorithm. The final point x_1^k returned by algorithm has the property that

$$E[F(x_t)] - F(x^*) \leq \frac{16G^2}{\lambda T} \tag{7}$$

Where the expectation is taken over the gradient estimates as well as the internal randomization of the algorithm.

Lemma

Define $V_k = \frac{G^2}{2^{k-2}\lambda}$, then for any k, we have $E[\triangle_k] \leq V_k$

High Probability Bounds

Theorem

Given $\delta > 0$ for success probability $1-\delta$, set $\tilde{\delta} = \frac{\delta}{k^\dagger}$ for $k^\dagger = \log(\frac{T}{450}+1)$. Set the parameter $T_1 = 450$, $\eta_1 = \frac{1}{3\lambda}$ and $D_1 = 2G\sqrt{\frac{\log(2/\tilde{\delta})}{\lambda}}$ in the EPOCH-GD-PROJ algorithm, The final point x_1^k returned by the algorithm has the property that with probability at least $1-\delta$, we have

$$F(x_1^k) - F(x^*) \leq \frac{1800G^2\log(2/\tilde{\delta})}{\lambda T}$$

The total number of gradient updates is at most T

Lemma used

Lemma

For any given $x^* \in \mathcal{K}$, let D be an upper bound on $||x_1 - x^*||$. Apply T iterations of the update

$$y_{t+1} = \nabla R^*(\nabla R(x_t) - \eta \hat{g}_t)$$
$$x_{t+1} = \underset{x \in \mathcal{K} \cap \mathcal{B}(x_1, D)}{argmin} \; B_R(x, y_{t+1})$$

where $\hat{g}_t$ is an unbiased estimator for the sub gradient of F at x_t satisfying strongly G-bound. Then for any $\delta \in (0,1)$, with probability at least $1 - \delta$ we have

$$\frac{1}{T}\sum_{t=1}^{T} F(x_t) - F(x^*) \leq \frac{\eta G^2 \log(2/\delta)}{2} + \frac{B_{\mathcal{R}}(x^*, x_1)}{\eta T} + \frac{4GD\sqrt{3\log(2/\delta)}}{\sqrt{T}}$$

By the convexity of F, the same bound also holds for $F(\bar{x}) - F(x^*)$, where $\bar{x} = \frac{1}{T}\sum_{t=1}^{T} x_t$

Theorem 2

For any online decision-making algorithm $\mathcal{A}$, there is a distribution over $\lambda-$strongly-convex cost functions with norms of gradients bounded by G such that

$$E[\text{Regret}(\mathcal{A})] = \Omega(\frac{G^2 \log(T)}{\lambda}) \tag{8}$$

Lemma

Let $p, p' \in [\frac{1}{4}, \frac{3}{4}]$ such that $|p' - p| \leq \frac{1}{8}$. Then

$$d_{TV}(B_p^n, B_{p'}^n) \leq \frac{1}{2}\sqrt{(p-p')^2 n}$$

where $d_{TV}(P, P') = \underset{A}{sup}|P(A) - P'(A)|$ and B_p^n: Bernoulli distribution on $\{0, 1\}$ with probability of obtaining 1 equal to p.

Lemma

Fix a round t. Let $\epsilon \leq \frac{1}{8\sqrt{t}}$ be a parameter. Let $p, p' \in [\frac{1}{4}, \frac{3}{4}]$ such that $2\epsilon \leq |p - p'| \leq 4\epsilon$. Then we have

$$E_p[Regret_t] + E_{p'}[Regret_t] \geq \frac{1}{4}\epsilon^2$$