COS-511: Learning Theory **Spring 2017**

Lecture 19: Strong Convexity & Second Order Methods

Lecturer: Roi Livni

 Logarithmic regret for

strongly convex funcitons We next revisit the OGD algorithm for special cases of convex function. Namely, we consider the OCO setting when the functions to be observed are *strongly convex*

Definition 19.1. *A convex function f is said to be α-strongly convex if*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{\alpha}{2}\|\mathbf{y} - \mathbf{x}\|^2 \tag{19.1}$$

19.0.1 OGD for strongly convex functions

We next, analyse the OGD algorithm for strongly convex functions

Theorem 19.2. *For α-strongly convex functions (and G-Lipschitz), OGD with step size $\eta_t = \frac{1}{\alpha t}$ achieves the following guarantee for all $T \geq 1$*

$$\text{Regret}_T \leq \frac{G^2}{2\alpha}(1 + \log T)$$

Proof. Define $\nabla_t = \nabla f(\mathbf{x}_t)$ and let $\mathbf{x}^* = \arg\min \sum_{t=1}^T f(\mathbf{x}^*)$. Applying the definition of strong convexity to the pair of points $\mathbf{x}^*, \mathbf{x}_t$ we have that

$$2f(\mathbf{x}_t) - f(\mathbf{x}^*) \geq 2\nabla f(\mathbf{x}_t)^\top (\mathbf{x}^* - \mathbf{x}) + \alpha\|\mathbf{x}^* - \mathbf{x}\|^2 \tag{19.2}$$

Using the update rule we have that

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 = \|\Pi_{\mathcal{K}}(\mathbf{x}_t - \eta_t \nabla_t) - \mathbf{x}^*\|^2 \leq \|\mathbf{x}_t - \eta_t \nabla_t - \mathbf{x}^*\|^2$$

Hence,

$$\|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2 - \|\mathbf{x}_t - \mathbf{x}^*\|^2 \leq \eta_t^2 \|\nabla_t\|^2 - 2\eta_t \nabla_t \cdot (\mathbf{x}_t - \mathbf{x}^*)$$

and,

$$2\nabla_t \cdot (\mathbf{x}_t - \mathbf{x}^*) \leq \frac{\|\mathbf{x}_t - \mathbf{x}^*\|^2 - \|\mathbf{x}_{t+1} - \mathbf{x}^*\|^2}{\eta_t} + \eta_t G^2 \tag{19.3}$$

Considering Eq. 19.2, Eq. 19.3 and summing up while considering $\eta_t = \frac{1}{\alpha t}$ (define $\frac{1}{\eta_0} = 0$):

$$\begin{aligned} & 2\sum_{t=1}^T f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*) \\ \leq & \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}^*\|^2 \left(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \alpha\right) + G^2 \sum_{t=1}^T \eta_t \\ = & 0 + G^2 \sum_{t=1}^T \frac{1}{\alpha t} \leq \frac{G^2}{\alpha}(1 + \log T) \end{aligned}$$

□

19.0.2 Learning Regulerized Objectives

We have so far applied SGD algorithm to optimize convex problems of the form

$$\begin{aligned} \text{minimize} \quad & \sum f_t(\mathbf{w}^*) \\ \text{s.t.} \quad & \|\mathbf{w}^*\| \leq B \end{aligned}$$

We've discussed that sometimes it is more convenient to consider unconstrained regulerized problems of the form

$$\text{minimize} \quad \frac{\lambda}{2}\|\mathbf{w}\|^2 + \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w})$$

The following claim is left as an exercise:

Claim 19.3. *For any convex function f: the funciton $\frac{\lambda}{2}\|\mathbf{w}\|^2 + f(\mathbf{w})$ is λ-strongly convex*

Theorem 19.4. *Apply OGD without projection (or with $B = \infty$), and set $\eta_t = \frac{1}{\lambda t}$ on a sequence of functions $\{\frac{\lambda}{2}\|\mathbf{w}\|^2 + f_t(\mathbf{w})\}_{t=1}^T$ and then for any $\|\mathbf{w}^*\| \leq B_0$ we have that:*

$$\sum_{t=1}^{T} f_t(\mathbf{w}_t) - f_t(\mathbf{w}^*) \leq \frac{G^2 \log T}{T\lambda} + \lambda B_0$$

Proof. First we bound the Lipschitness of the functions $\frac{\lambda}{2}\|\mathbf{w}\|^2 + f_t(\mathbf{w})$:

Note that $\nabla_t = \lambda \mathbf{w}_t + \nabla f_t(\mathbf{w}_t)$, hence

$$\begin{aligned} \mathbf{w}_{t+1} =& \mathbf{w}_t - \frac{1}{\lambda t}\nabla_t \\ &= (1 - \frac{1}{t})\mathbf{w}_t + \frac{1}{t}\left(\frac{1}{\lambda}\nabla f_t(\mathbf{w}_t)\right) \end{aligned}$$

Hence we have that:

$$\|\mathbf{w}_{t+1}\| = \|\frac{1}{t}\sum \frac{1}{\lambda}\nabla f_t(\mathbf{w}_t)\| \leq \frac{G}{\lambda}$$

We thus have that

$$\|\nabla_t\| \leq 2G$$

Thus, the result follows from applying the regret bound for strongly convex functions we get that for any $\mathbf{w}^*$:

$$\sum_{t=1}^{T} \frac{\lambda}{2}\|\mathbf{w}_t\|^2 + f_t(\mathbf{w}_t) - \frac{\lambda}{2}\|\mathbf{w}^*\| + f_t(\mathbf{w}^*) \leq \frac{4G^2 \log T}{2\lambda}$$

Hence

$$\frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}_t) - f_t(\mathbf{w}^*) \leq \frac{2G^2 \log T}{T\lambda} + \frac{\lambda}{2}B^2$$

□

Applying the Online to Batch theorem we can thus obtain an algorithm for learning regulerized objective. Note that if we want to achieve $\frac{1}{\sqrt{T}}$ error, we would need to set $\lambda = O(\frac{1}{\sqrt{T}})$ which would still have the same overall learning rate, but this algorithm avoids making projection steps.

19.1 Second Order Methods

19.1.1 Motivation – Online Portfolio Selection

To motivate the construction of second order method, we return to the problem of online portfolio selection. In online portofolio selection, at each iteration the learner chooses to distribute her wealther amongst n stocks. Similarly to the online expert model her decision is a vector $\mathbf{x}_t \in \Delta_n$. Nature then chooses a vector $\|\mathbf{r}_t\|_\infty \leq 1$, where $\mathbf{r}_t(i) = \frac{\text{price of stock i at time } t+1}{\text{price of stock i at time } t}$. For exposition, will assume for simplicity that $\min_i \mathbf{r}_t(i) \geq \frac{1}{G}$ (i.e. no stock ever loses its whole value). The revenue of the learner is measure in terms of $\log \mathbf{x}_t \cdot \mathbf{r}_t$. The overall gain of the learner is then

$$\sum \log \mathbf{x}_t \cdot \mathbf{r}_t = \log \prod \mathbf{x}_t \cdot \mathbf{r}_t = \log \frac{\text{Wealth at time T}}{\text{Wealth at time 1}}.$$

The regret is measure against the best constant rebalance portfolio (such a portfolio strategy is called *universal*)

$$\text{Regret}_T = \max_{\mathbf{x}^* \in \Delta_n} \sum \log(\mathbf{x}^* \cdot \mathbf{r}_t) - \sum \log(\mathbf{x}_t \cdot \mathbf{r}_t)$$

Note that we have a maximization problem and not a minimization problem, yet log is a concave and not convex problem therefore the problem is equivalent to

$$\text{Regret}_T = \sum -\log(\mathbf{x}_t \cdot \mathbf{r}_t) - \min_{\mathbf{x}^* \in \Delta_n} \sum -\log(\mathbf{x}^* \cdot \mathbf{r}_t)$$

As before we can apply OGD to obtain $O(\sqrt{T})$ regret (Note that our assumption that $\mathbf{r}_t \geq \frac{1}{G}$ implies that the loss vectors are always Lipschitz). However, the log function is in fact, an *exp−−concave* function which allows us to improve on these results using second order methods.

19.1.2 Second Order Methods

The idea behind second order methods is that if gradient descent *linearizes* the function and chooses a step according to the first order approximation, a second order method consider always the *Hessian* of the function (i.e. its second derivative). For example the Hessian of the $\log(\mathbf{r}_t \cdot \mathbf{x}_t)$ function is given by

$$\nabla^2 f_t(\mathbf{x}) = \frac{\mathbf{r}_t \cdot \mathbf{r}_t^\top}{(\mathbf{r}_t \cdot \mathbf{x})^2},$$

and it is the derivative of the gradient function. The crucial property that we wish to exploit, is that the Hessian of a the function is large in the direction of the gradient:

Definition 19.5. *A convex function f is called $\alpha - \exp$-concave over $\mathcal{K}$ if the function g is concave, where g is defined as*

$$g(\mathbf{x}) = e^{-\alpha f(\mathbf{x})}.$$

The following Lemma is left as an exercise:

Lemma 19.6. *A twice-differentiable function f is $\alpha - \exp$ concave if and only if*

$$\nabla^2 f \succ \alpha \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^\top$$

In other words $\nabla^2 f - \alpha \nabla f(\mathbf{x}) \nabla f(\mathbf{x})^\top$ is a p.s.d matrix.

19.1.3 Online Newton Step

We next describe the online newton step algorithm. For that we add one more notation, we will denote by $\Pi_{\mathcal{K}}^{A}$ the projection over the set $\mathcal{K}$, with respect to the metric A: i.e.

$$\Pi_{\mathcal{K}}^{A}(\mathbf{y}) = \arg\min_{\mathbf{x}\in\mathcal{K}}(\mathbf{x}-\mathbf{y})A(\mathbf{x}-\mathbf{y})$$

Algorithm 1 Online Newton Step ONS

Inititalization $\mathbf{x}_1 \in \mathcal{K}$, parameters $\gamma, \epsilon > 0$, $A_0 = \epsilon\mathbf{Id}$.
for $t = 1, 2 \dots T$ **do**
Play $\mathbf{x}_t$ and observe cost $f_t(\mathbf{x}_t)$
Rank 1 update $A_t = A_{t-1} + \nabla_t \nabla_t^\top$.
Newton step and projection:

$$\mathbf{y}_{t+1} = \mathbf{x}_t - \frac{1}{\gamma}A_t^{-1}\nabla_t$$
$$\mathbf{x}_{t+1} = \Pi_{\mathcal{K}}^{A_t}(\mathbf{y}_{t+1}$$

end for
return

Theorem 19.7. *Alg. 1 with parameters* $\gamma = \min\{\frac{1}{4GD}, \alpha\}$ *and* $\epsilon = \frac{1}{\gamma^2 D^2}$*, guarantees (for* $T > 4$*):*

$$\text{Regret}_T \leq 5(\frac{1}{\alpha} + GD)n\log T$$