E0 370 Statistical Learning Theory	**Lecture 19 (Oct 22, 2013)**

Online Convex Optimization

Lecturer: Shivani Agarwal | *Scribe: Aadirupa*

1 Introduction

In this lecture we shall look at a fairly general setting of online convex optimization which, as we shall see, encompasses some of the online learning problems we have seen so far as special cases. A standard (offline) convex optimization problem involves a convex set $\Omega \subseteq \mathbb{R}^n$ and a fixed convex cost function $c : \Omega \mapsto \mathbb{R}$; the goal is to find a point $\mathbf{x}^* \in \Omega$ that minimizes $c(\mathbf{x})$ over Ω. An online convex optimization problem also involves a convex set $\Omega \subseteq \mathbb{R}^n$, but proceeds in trials: at each trial t, the player/algorithm must choose a point $\mathbf{x}^t \in \Omega$, after which a convex cost function $c_t : \Omega \to \mathbb{R}$ is revealed by the environment/adversary, and a loss of $c_t(\mathbf{x}^t)$ is incurred for the trial; the goal is to minimize the total loss incurred over a sequence of trials, relative to the best single point in Ω in hindsight.

Online Convex Optimization

Convex set $\Omega \subseteq \mathbb{R}^n$
For $t = 1, \ldots, T$:
 – Play $\mathbf{x}^t \in \Omega$
 – Receive cost function $c_t : \Omega \to \mathbb{R}$
 – Incur loss $c_t(\mathbf{x}^t)$

We will denote the total loss incurred by an algorithm $\mathcal{A}$ on T trials as

$$L_T[\mathcal{A}] = \sum_{t=1}^{T} c_t(\mathbf{x}^t)\,,$$

and the total loss of a fixed point $\mathbf{x} \in \Omega$ as

$$L_T[\mathbf{x}] = \sum_{t=1}^{T} c_t(\mathbf{x})\,;$$

the regret of $\mathcal{A}$ w.r.t. the best fixed point in Ω is then simply

$$L_T[\mathcal{A}] - \inf_{\mathbf{x}\in\Omega} L_T[\mathbf{x}]\,.$$

Let us consider which of the online learning problems we have seen earlier can be viewed as instances of online convex optimization (OCO):

1. **Online linear regression (GD analysis).** Here one selects a weight vector $\mathbf{w}^t \in \mathbb{R}^n$, and incurs a loss given by $c_t(\mathbf{w}^t) = \ell_{y^t}(\mathbf{w}^t \cdot \mathbf{x}^t)$, which is assumed to be convex in its argument, and is therefore convex in $\mathbf{w}^t$; however the total loss is compared against the best fixed weight vector in only $\{\mathbf{u} \in \mathbb{R}^n : \|\mathbf{u}\|_2 \le U\}$ for some $U > 0$. Call this latter set Ω. Clearly, if the weight vectors $\mathbf{w}^t$ are also selected to be in Ω, e.g. by Euclidean projection onto Ω (which in this case amounts to a simple normalization), then this becomes an instance of OCO.

2. **Online linear regression (EG analysis).** Here one selects $\mathbf{w}^t \in \Delta_n$, and incurs a loss given by $c_t(\mathbf{w}^t) = \ell_{y^t}(\mathbf{w}^t \cdot \mathbf{x}^t)$, which is assumed to be convex in its argument, and is therefore convex in $\mathbf{w}^t$; the total loss is also compared against the best fixed weight vector in Δ_n. Thus this can be viewed as an instance of OCO with $\Omega = \Delta_n$.

3. **Online prediction from expert advice.** Here one selects a probability vector $\hat{\mathbf{p}}^t \in \Delta_n$, and incurs a loss $c_t(\hat{\mathbf{p}}^t) = \ell_{y^t}(\hat{\mathbf{p}}^t \cdot \xi^t)$, which is assumed to be convex in its argument, and is therefore convex in $\hat{\mathbf{p}}^t$. However the total loss is compared only against the best vector in $\big\{\mathbf{e}_i \in \Delta_n : i \in [n]\big\}$, where $\mathbf{e}_i$ denotes a unit vector with 1 in the i-th position and 0 elsewhere. In general, the smallest loss in this set is not the same as the smallest loss in Δ_n, and therefore this is *not* an instance of OCO.

4. **Online decision/allocation among experts.** Here one selects a probability vector $\hat{\mathbf{p}}^t \in \Delta_n$, and incurs a *linear* loss $c_t(\hat{\mathbf{p}}^t) = \hat{\mathbf{p}}^t \cdot \boldsymbol{\ell}^t$, which is clearly convex in $\hat{\mathbf{p}}^t$. The total loss is again compared against the best vector in $\big\{\mathbf{e}_i \in \Delta_n : i \in [n]\big\}$ as above, but in this case, comparison against this set is actually equivalent to comparison against Δ_n, since

$$\inf_{\hat{\mathbf{p}} \in \Delta_n} L_T[\hat{\mathbf{p}}] = \inf_{\hat{\mathbf{p}} \in \Delta_n} \sum_{t=1}^T \hat{\mathbf{p}} \cdot \boldsymbol{\ell}_t = \inf_{\hat{\mathbf{p}} \in \Delta_n} \hat{\mathbf{p}} \cdot \Big(\sum_{t=1}^T \boldsymbol{\ell}_t \Big) = \min_{i \in [n]} \mathbf{e}_i \cdot \Big(\sum_{t=1}^T \boldsymbol{\ell}_t \Big) = \min_{i \in [n]} \sum_{t=1}^T \mathbf{e}_i \cdot \boldsymbol{\ell}_t \,.$$

Thus online allocation with a linear loss as above can be viewed as an instance of OCO with $\Omega = \Delta_n$.

2 Online Gradient Descent Algorithm

The online gradient descent (OGD) algorithm [7] generalizes the basic gradient descent algorithm used for standard (offline) optimization problems, and can be described as follows:

Algorithm **Online Gradient Descent (OGD)**

Parameter: $\eta > 0$
Initialize: $\mathbf{x}^1 \in \Omega$
For $t = 1, \ldots, T$:
– Play $\mathbf{x}^t \in \Omega$
– Receive cost function $c_t : \Omega \mapsto \mathbb{R}$
– Incur loss $c_t(\mathbf{x}^t)$
– Update:
$\widetilde{\mathbf{x}}^{t+1} \leftarrow \mathbf{x}^t - \eta \nabla c_t(\mathbf{x}^t)$
$\mathbf{x}^{t+1} \leftarrow P_\Omega(\widetilde{\mathbf{x}}^{t+1})$,
where $P_\Omega(\widetilde{\mathbf{x}}) = \operatorname{argmin}_{\mathbf{x} \in \Omega} \|\mathbf{x} - \widetilde{\mathbf{x}}\|_2$ (Euclidean projection of $\widetilde{\mathbf{x}}$ onto Ω)

Theorem 2.1 (Zinkevich, 2003)**.** Let $\Omega \subseteq \mathbb{R}^n$ be a closed, non-empty, convex set with $\|\mathbf{x} - \mathbf{x}^1\|_2 \leq D \;\forall \mathbf{x} \in \Omega$. Let $c_t : \Omega \to \mathbb{R}$ $(t = 1, 2, \ldots)$ be any sequence of convex cost functions with bounded subgradients, $\|\nabla c_t(\mathbf{x})\|_2 \leq G \;\forall t, \;\forall \mathbf{x} \in \Omega$. Then

$$L_T\big[\mathrm{OGD}(\eta)\big] - \inf_{\mathbf{x} \in \Omega} L_T[\mathbf{x}] \;\leq\; \frac{1}{2} \left(\frac{D^2}{\eta} + \eta \, G^2 T \right) .$$

Moreover, setting $\eta^* = \frac{D}{G\sqrt{T}}$ to minimize the right-hand side of the above bound (assuming D, G and T, or upper bounds on them, are known in advance), we have

$$L_T\big[\mathrm{OGD}(\eta^*)\big] - \inf_{\mathbf{x} \in \Omega} L_T[\mathbf{x}] \;\leq\; DG\sqrt{T} \,.$$

Notes. Before proving the above theorem, we note the following:

1. The analysis of the GD algorithm for online regression can be viewed as a special case of the above, with $D \leq U$ and $G \leq ZR_2$ (see Lecture 17 notes).

2. With η^* as above (which depends on knowledge of D, G and T), one gets an $O(\sqrt{T})$ regret bound. Such an $O(\sqrt{T})$ regret bound (with worse constants) can also be obtained by using a time-varying parameter $\eta_t = 1/\sqrt{t} \;\forall t$ (in fact this was the version contained in the original paper of Zinkevich [7]).

Proof of Theorem 2.1. The proof is similar to that of the regret bound we obtained for the GD algorithm for online regression in Lecture 17. Let $\mathbf{x} \in \Omega$. We have $\forall t \in [T]$:

$$\begin{aligned}
\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) &\leq \nabla c_t(\mathbf{x}^t) \cdot (\mathbf{x}^t - \mathbf{x}), \quad \text{by convexity of } c_t \\
&= \left(\frac{\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}}{\eta}\right) \cdot (\mathbf{x}^t - \mathbf{x}) \\
&= \frac{1}{2\eta}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \widetilde{\mathbf{x}}^{t+1}\|_2^2 + \|\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}\|_2^2\Big) \\
&\leq \frac{1}{2\eta}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \mathbf{x}^{t+1}\|_2^2 + \|\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}\|_2^2\Big) \\
&= \frac{1}{2\eta}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \mathbf{x}^{t+1}\|_2^2 + \eta^2\|\nabla c_t(\mathbf{x}^t)\|_2^2\Big)
\end{aligned}$$

Summing over $t = 1 \ldots T$, we thus get

$$\begin{aligned}
\sum_{t=1}^{T}\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) &\leq \frac{1}{2\eta}\Big(\|\mathbf{x} - \mathbf{x}^1\|_2^2 - \|\mathbf{x} - \mathbf{x}^{T+1}\|_2^2 + \eta^2 G^2 T\Big) \\
&\leq \frac{1}{2\eta}\big(D^2 + \eta^2 G^2 T\big).
\end{aligned}$$

It is easy to see that the right-hand side is minimized at $\eta^* = \frac{D}{G\sqrt{T}}$; substituting this back in the above inequality gives the desired result. □

2.1 Logarithmic regret bound for OGD when cost functions are strongly convex (using time-varying parameter η_t)

The above result gives an $O(\sqrt{T})$ regret bound for the OGD algorithm in general. When the cost functions c_t are known to be α-strongly convex, one can in fact obtain an $O(\ln T)$ regret bound with a slight variant of the OGD algorithm that uses a suitable time-varying parameter η_t [3]:

Theorem 2.2 (Hazan et al, 2007)**.** Let $\Omega \subseteq \mathbb{R}^n$ be a closed, non-empty, convex set with $\|\mathbf{x} - \mathbf{x}^1\|_2 \leq D \ \forall \mathbf{x} \in \Omega$. Let $c_t : \Omega \to \mathbb{R}$ $(t = 1, 2, \ldots)$ be any sequence of α-strongly convex cost functions with bounded subgradients, $\|\nabla c_t(\mathbf{x}^t)\|_2 \leq G \ \forall t, \ \forall \mathbf{x} \in \Omega$. Then the OGD algorithm with time-varying parameter $\eta_t = \frac{1}{\alpha t}$ satisfies

$$L_T\left[OGD\left(\eta_t = \frac{1}{\alpha t}\right)\right] - \inf_{\mathbf{x} \in \Omega} L_T[\mathbf{x}] \ \leq \ \frac{G^2}{2\alpha}\big(\ln T + 1\big).$$

Proof. Let $\mathbf{x} \in \Omega$. Proceeding similarly as before, we have $\forall t \in [T]$:

$$\begin{aligned}
\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) &\leq \nabla c_t(\mathbf{x}^t) \cdot (\mathbf{x}^t - \mathbf{x}) - \frac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}\|_2^2, \quad \text{by } \alpha\text{-strong convexity of } c_t \\
&= \left(\frac{\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}}{\eta_t}\right) \cdot (\mathbf{x}^t - \mathbf{x}) - \frac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}\|_2^2 \\
&= \frac{1}{2\eta_t}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \widetilde{\mathbf{x}}^{t+1}\|_2^2 + \|\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}\|_2^2 - \alpha\eta_t\|\mathbf{x}^t - \mathbf{x}\|_2^2\Big) \\
&\leq \frac{1}{2\eta_t}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \mathbf{x}^{t+1}\|_2^2 + \|\mathbf{x}^t - \widetilde{\mathbf{x}}^{t+1}\|_2^2 - \alpha\eta_t\|\mathbf{x}^t - \mathbf{x}\|_2^2\Big) \\
&= \frac{1}{2\eta_t}\Big(\|\mathbf{x} - \mathbf{x}^t\|_2^2 - \|\mathbf{x} - \mathbf{x}^{t+1}\|_2^2 + {\eta_t}^2\|\nabla c_t(\mathbf{x}^t)\|_2^2 - \alpha\eta_t\|\mathbf{x}^t - \mathbf{x}\|_2^2\Big).
\end{aligned}$$

Summing over $t = 1 \ldots T$ then gives

$$\begin{aligned}
\sum_{t=1}^{T}\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) &\leq \frac{G^2}{2}\sum_{t=1}^{T}\eta_t + \frac{1}{2}\Big(\frac{1}{\eta_1} - \alpha\Big)\|\mathbf{x} - \mathbf{x}^1\|_2^2 + \frac{1}{2}\sum_{t=2}^{T}\Big(\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \alpha\Big)\|x - x^t\|_2^2 \\
&\leq \frac{G^2}{2\alpha}\big(1 + \ln T\big) + 0 + 0.
\end{aligned}$$

□

3 Online Mirror Descent Algorithm

The online mirror descent (OMD) algorithm, described for example in [6], generalizes the basic mirror descent algorithm used for standard (offline) optimization problems [4, 5, 1] much as OGD generalizes basic gradient descent. The OMD framework allows one to obtain $O(\sqrt{T})$ regret bounds for cost functions with bounded subgradients not only in Euclidean norm (assumed by the OGD analysis), but in any suitable norm.

Let $\widetilde{\Omega} \in \mathbb{R}^n$ be a convex set containing Ω (i.e. $\Omega \subseteq \widetilde{\Omega}$), and let $F : \widetilde{\Omega} \to \mathbb{R}$ be a strictly convex and differentiable function. Let $B_F : \widetilde{\Omega} \times \widetilde{\Omega} \to \mathbb{R}$ denote the Bregman divergence associated with F; recall that for all $\mathbf{u}, \mathbf{v} \in \widetilde{\Omega}$, this is defined as

$$B_F(\mathbf{u}, \mathbf{v}) = F(\mathbf{u}) - F(\mathbf{v}) - (\mathbf{u} - \mathbf{v}) \cdot \nabla F(\mathbf{v}) \,.$$

Then the OMD algorithm using the set $\widetilde{\Omega}$ and function F can be described as follows:

Algorithm **Online Mirror Descent (OMD)**

Parameters: $\eta > 0$; convex set $\widetilde{\Omega} \supseteq \Omega$; strictly convex, differentiable function $F : \widetilde{\Omega} \to \mathbb{R}$
Initialize: $\mathbf{x}^1 = \operatorname*{argmin}_{\mathbf{x} \in \Omega} F(\mathbf{x})$
For $t = 1, \ldots, T$:
– Play $\mathbf{x}^t \in \Omega$
– Receive cost function $c_t : \Omega \to \mathbb{R}$
– Incur loss $c_t(\mathbf{x}^t)$
– Update:
$\nabla F(\widetilde{\mathbf{x}}^{t+1}) \leftarrow \nabla F(\mathbf{x}^t) - \eta \nabla c_t(\mathbf{x}^t)$ (Assume this yields $\widetilde{\mathbf{x}}^{t+1} \in \widetilde{\Omega}$)
$\mathbf{x}^{t+1} \leftarrow P_\Omega^F(\widetilde{\mathbf{x}}^{t+1})$,
where $P_\Omega^F(\widetilde{\mathbf{x}}) = \operatorname*{argmin}_{\mathbf{x} \in \Omega} B_F(\mathbf{x}, \widetilde{\mathbf{x}})$ (Bregman projection of $\widetilde{\mathbf{x}}$ onto Ω)

Below we will obtain a regret bound for OMD (with appropriate choice of the function F) for cost functions with subgradients bounded in an arbitrary norm. Before we do so, let us recall the notion of a *dual norm.* Specifically, let $\|\cdot\|$ be any norm defined on a closed convex set $\widetilde{\Omega} \subseteq \mathbb{R}^n$. Then the corresponding dual norm is defined as

$$\|\mathbf{u}\|_* \;=\; \sup_{\mathbf{x} \in \widetilde{\Omega} : \|\mathbf{x}\| \le 1} \mathbf{x} \cdot \mathbf{u} \;=\; 2 \sup_{\mathbf{x} \in \widetilde{\Omega}} \Big(\mathbf{x} \cdot \mathbf{u} - \tfrac{1}{2} \|\mathbf{x}\|^2 \Big) \,.$$

Theorem 3.1. Let $\Omega \subseteq \mathbb{R}^n$ be a closed, non-empty, convex set. Let $c_t : \Omega \to \mathbb{R}$ $(t = 1, 2, \ldots)$ be any sequence of convex cost functions with bounded subgradients $\|\nabla c_t(\mathbf{x}^t)\| \le G \;\forall t,\; \forall x \in \Omega$ (where $\|\cdot\|$ is an arbitrary norm). Let $\widetilde{\Omega} \supseteq \Omega$ be a convex set and $F : \widetilde{\Omega} \to \mathbb{R}$ be a strictly convex function such that

(a) $F(\mathbf{x}) - F(\mathbf{x}^1) \le D^2 \;\forall \mathbf{x} \in \Omega$; and

(b) the restriction of F to Ω is α-strongly convex w.r.t. $\|\cdot\|_*$, the dual norm of $\|\cdot\|$.

Then

$$L_T\big[\mathrm{OMD}(\eta)\big] - \inf_{\mathbf{x} \in \Omega} L_T[\mathbf{x}] \;\le\; \frac{1}{\eta} \left(D^2 + \frac{\eta^2 G^2 T}{2\alpha} \right) .$$

Moreover, setting $\eta^* = \frac{D}{G}\sqrt{\frac{2\alpha}{T}}$ to minimize the right-hand side of the above bound (assuming D, G, T and α, or upper bounds on them, are known in advance), we have

$$L_T\big[\mathrm{OMD}(\eta^*)\big] - \inf_{\mathbf{x} \in \Omega} L_T[\mathbf{x}] \;\le\; DG\sqrt{\frac{2T}{\alpha}} \,.$$

Proof. Let $\mathbf{x} \in \Omega$. Proceeding as before, we have $\forall t \in [T]$:

$$\begin{aligned}
\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) &\leq \nabla c_t(\mathbf{x}^t) \cdot (\mathbf{x}^t - \mathbf{x})\,, \quad \text{by convexity of } c_t \\
&= \left(\frac{\nabla F(\mathbf{x}^t) - \nabla F(\widetilde{\mathbf{x}}^{t+1})}{\eta}\right) \cdot (\mathbf{x}^t - \mathbf{x}) \\
&= \frac{1}{\eta}\Big(B_F(\mathbf{x}, \mathbf{x}^t) - B_F(\mathbf{x}, \widetilde{\mathbf{x}}^{t+1}) + B_F(\mathbf{x}^t, \widetilde{\mathbf{x}}^{t+1})\Big) \\
&\leq \frac{1}{\eta}\Big(B_F(\mathbf{x}, \mathbf{x}^t) - B_F(\mathbf{x}, \mathbf{x}^{t+1}) - B_F(\mathbf{x}^{t+1}, \widetilde{\mathbf{x}}^{t+1}) + B_F(\mathbf{x}^t, \widetilde{\mathbf{x}}^{t+1})\Big)\,.
\end{aligned}$$

Summing over $t = 1 \ldots T$, we thus get

$$\sum_{t=1}^{T} \big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) \;\leq\; \frac{1}{\eta}\Big(B_F(\mathbf{x}, \mathbf{x}^1) - B_F(\mathbf{x}, \mathbf{x}^{T+1})\Big) + \frac{1}{\eta}\sum_{t=1}^{T}\Big(B_F(\mathbf{x}^t, \widetilde{\mathbf{x}}^{t+1}) - B_F\left(\mathbf{x}^{t+1}, \widetilde{\mathbf{x}}^{t+1}\right)\Big)\,.$$

Now, it can be shown that

$$B_F(\mathbf{x}, \mathbf{x}^1) \;\leq\; F(\mathbf{x}) - F(\mathbf{x}^1) \;\leq\; D^2\,.$$

Also, we have

$$\begin{aligned}
&\sum_{t=1}^{T}\Big(B_F(\mathbf{x}^t, \widetilde{\mathbf{x}}^{t+1}) - B_F(\mathbf{x}^{t+1}, \widetilde{\mathbf{x}}^{t+1})\Big) \\
&= \sum_{t=1}^{T}\Big(F(\mathbf{x}^t) - F(\mathbf{x}^{t+1}) - \nabla F(\widetilde{\mathbf{x}}^{t+1}) \cdot (\mathbf{x}^t - \mathbf{x}^{t+1})\Big) \\
&\leq \sum_{t=1}^{T}\bigg(\Big(\nabla F(\mathbf{x}^t) \cdot (\mathbf{x}^t - \mathbf{x}^{t+1}) - \frac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_*^2\Big) - \nabla F(\widetilde{x}^{t+1}).(x^t - x^{t+1})\bigg)\,, \\
&\qquad \text{by } \alpha\text{-strong convexity of } F \text{ on } \Omega \\
&= \sum_{t=1}^{T}\Big(-\eta \nabla c_t(\mathbf{x}^t) \cdot (\mathbf{x}^t - \mathbf{x}^{t+1}) - \frac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_*^2\Big) \\
&\leq \sum_{t=1}^{T}\Big(\eta G\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_* - \frac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_*^2\Big)\,, \quad \text{by Hölders inequality} \\
&\leq \sum_{t=1}^{T}\frac{\eta^2 G^2}{2\alpha}\,, \\
&\qquad \text{since } \tfrac{\eta^2 G^2}{2\alpha} + \tfrac{\alpha}{2}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_*^2 - \eta G\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_* = \Big(\tfrac{\eta G}{\sqrt{2\alpha}} - \sqrt{\tfrac{\alpha}{2}}\|\mathbf{x}^t - \mathbf{x}^{t+1}\|_*\Big)^2 \geq 0 \\
&= \frac{\eta^2 G^2 T}{2\alpha}\,.
\end{aligned}$$

Thus we get

$$\sum_{t=1}^{T}\big(c_t(\mathbf{x}^t) - c_t(\mathbf{x})\big) \;\leq\; \frac{1}{\eta}\left(D^2 + \frac{\eta^2 G^2 T}{2\alpha}\right)\,.$$

It is easy to verify that the right-hand side is minimized at $\eta^* = \frac{D}{G}\sqrt{\frac{2\alpha}{T}}$; substituting this back in the above inequality gives the desired result. $\square$

Examples. Examples of OMD algorithms include the following:

1. **OGD.** Here $\widetilde{\Omega} = \mathbb{R}^n$ and $F(\mathbf{x}) = \frac{1}{2}\|\mathbf{x}\|_2^2$. Note that F is 1-strongly convex on $\mathbb{R}^n$ (and therefore on any $\Omega \subseteq \mathbb{R}^n$).
2. **Exponential weights.** Here $\Omega = a\Delta_n$ for some $a > 0$; $\widetilde{\Omega} = \mathbb{R}^n_{++}$; and $F(\mathbf{x}) = \sum_{i=1}^n x_i \ln x - \sum_{i=1}^n x_i$. Note that F is strictly convex on $\widetilde{\Omega}$ and 1-strongly convex on $\Omega = \Delta_n$.

Exercise. Suppose the cost functions c_t are all β-strongly convex for some $\beta > 0$ (and have subgradients bounded in some arbitrary norm $\|\cdot\|$ as above). Can you obtain a logarithmic ($O(\ln(T))$) regret bound for the OMD algorithm in this case, as we did for the OGD algorithm? Why or why not?

More details on online convex optimization, including missing details in some of the above proofs, can be found for example in [2].

4 Next Lecture

In the next lecture, we will consider how algorithms for online supervised learning problems (such as online regression and classification) can be used in a batch setting, and will see how regret bounds for online learning algorithms can be converted to generalization error bounds for the resulting batch algorithms.

References

[1] Amir Beck and Marc Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

[2] Sébastion Bubeck. Introduction to online optimization. Lecture Notes, Princeton University, 2011.

[3] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69:169–192, 2007.

[4] A. Nemirovski. Efficient methods for large-scale convex optimization problems. *Ekonomika i Matematicheskie Metody*, 15, 1979.

[5] A. Nemirovski and D. Yudin. *Problem Complexity and Method Efficiency in Optimization*. Wiley Interscience, 1983.

[6] Shai Shalev-Shwartz. *Online Learning: Theory, Algorithms, and Applications*. PhD thesis, The Hebrew University of Jerusalem, 2007.

[7] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning (ICML)*, 2003.