Lecture 27: Online Convex Optimization and Mirror Descent

Yudong Chen

Reading:
- Chapter 21 of Duchi’s notes.
- Xinhua Zhang, short notes on mirror descent,
- Elad Hazan, “Introduction to Online Convex Optimization”.

1 Online Convex Optimization

The setup can be described as a two-player sequential game:
- Let $\mathcal{X} \subseteq \mathbb{R}^d$ be a convex parameter space.
- At each time $t$, player 1 (the learner) chooses some $x_t \in \mathcal{X}$.
- Player 2 (the adversary, or nature) then chooses a loss function $f_t : \mathcal{X} \to \mathbb{R}$, where $f_t$ is convex.

Note that the learner commits to $x_t$ before seeing $f_t$, whereas the adversary may adapt its choice of $f_t$ to $x_1, \ldots, x_t$. The goal for the learner is to minimize the average regret (or optimality gap), defined as

$$\frac{1}{T} \sum_{t=1}^{T} (f_t(x_t) - f_t(x^*)),$$

where $x^* := \arg\min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x)$ is the best fixed decision in hindsight.

1.1 Examples

Here are some examples of problems that fall into the framework of online convex optimization.

1. **Online support vector machine**: At each time $t$, the learner picks a vector $x_t \in \mathbb{R}^d$. Then, a data point $(a_t, y_t) \in \mathbb{R}^d \times \{\pm 1\}$ is revealed, and the learner incurs loss $f_t(x_t)$, where $f_t(x) = \max\{1 - y_t \langle x, a_t \rangle, 0\}$. (This loss function is called the hinge loss.)

2. **Online logistic regression**: Same setup, except now the loss function is $f_t(x) = \log(1 + e^{-y_t \langle x, a_t \rangle})$. (This is the logistic loss.)

3. **Expert prediction/adversarial bandit**: There are $d$ experts/arms. At each time $t$, each expert makes a prediction (for example “I predict the stock market will go up tomorrow”). At each time $t$, the learner chooses a weight vector $x_t = (x_{t1}, \ldots, x_{td})$, where

$$x_{tj} = \text{weight for expert } j = \text{probability of pulling arm } j.$$
So the parameter space is \( \mathcal{X} = \Delta_d := \{ x \in \mathbb{R}^d : \sum_j x_j = 1, x_j \geq 0 \} \), which is the probability simplex in \( \mathbb{R}^d \). Then losses
\[
l_{ij} = \mathbb{I}\{ \text{expert } j \text{ is wrong at time } t \} = \text{loss of arm } j \text{ at time } t
\]
are revealed, and the learner incurs loss \( f_t(x) = \langle x, l_t \rangle \). Note that \( \nabla f_t(x) = l_t \).

## 2 Online Gradient Descent

Gradient descent extends naturally to an algorithm for online convex optimization. Online gradient descent does, at each iteration \( t + 1 \):
\[
x_{t+1} = P_{\Delta_d} (x_t - \alpha_t g_t)
\]
\[
= \arg\min_{x \in \Delta_d} \left\{ \langle g_t, x \rangle + \frac{1}{2\alpha_t} \| x - x_t \|_2^2 \right\}.
\]
where \( \alpha_t \) is the step size and \( g_t \in \partial f_t(x_t) \) is a subgradient of \( f_t \) at \( x_t \). (If \( f_t \) is differentiable, then \( g_t = \nabla f_t(x_t) \).)

## 3 Bregman Divergence

We will next see how to extend gradient descent to a more general algorithm. First, we will need to introduce the notion of Bregman divergence. Let \( \psi : \mathbb{R}^d \to \mathbb{R} \) be a differentiable convex function.

**Definition 1** (Bregman Divergence). The **Bregman divergence** associated with \( \psi \) is a function \( B_{\psi} : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R} \) defined by
\[
B_{\psi}(x, y) := \psi(x) - \psi(y) - \langle \nabla \psi(y), x - y \rangle
\]

**Remark 1.** By the convexity of \( \psi \), the Bregman divergence \( B_{\psi} \) is always non-negative. One can think of \( B_{\psi}(x, y) \) as a measure of “distance” between \( x \) and \( y \); however, the Bregman divergence is not necessarily symmetric or satisfies the triangle inequality.

### 3.1 Examples

1. **Euclidean distance.** Let \( \psi(x) = \frac{1}{2} \| x \|_2^2 \). Then \( B_{\psi}(x, y) = \frac{1}{2} \| x - y \|_2^2 \).

2. **Mahalanobis distance.** Let \( \psi(x) = \frac{1}{2} x^\top A x =: \frac{1}{2} \| x \|_A^2 \), where \( A \succeq 0 \).

   Then \( B_{\psi}(x, y) = \frac{1}{2} (x - y)^\top A (x - y) = \frac{1}{2} \| x - y \|_A^2 \).

3. **KL-divergence.** Let \( \psi(x) = \sum_{j=1}^d x_j \log x_j \) be the negative entropy. Note that \( \psi \) is convex on \( \mathbb{R}_+^d \).

   Then \( B_{\psi}(x, y) = \sum_{j=1}^d x_j \log \frac{x_j}{y_j} = D_{\text{KL}}(x, y) \) for all \( x, y \in \Delta_d \).
4 Online Mirror Descent (OMD)

This is a generalization of gradient descent using Bregman divergences. At iteration $t$:

$$x_{t+1} = \arg\min_{x \in \mathcal{X}} \left\{ \langle g_t, x \rangle + \frac{1}{\alpha_t} B_{\psi}(x, x_t) \right\}$$  \hspace{1cm} (1)

**Remark 2.** $\langle g_t, x \rangle + \frac{1}{\alpha_t} B_{\psi}(x, x_t)$ is convex in $x$. Hence this is a convex optimization problem.

4.1 Special cases of OMD

**Gradient descent** $\psi(x) = \frac{1}{2} \|x\|_2^2$

**Exponentiated gradient descent** This is online mirror descent with $\mathcal{X} = \Delta_d$, $\psi(x) = \sum_j x_j \log x_j$, and $B_{\psi}(x, y) = D_{\text{KL}}(x, y)$. At iteration $t$:

$$x_{t+1} = \arg\min_{x \in \mathcal{X}} \left\{ \langle g, x \rangle + \frac{1}{\alpha_t} D_{\text{KL}}(x, x_t) \right\}.$$

To explicit calculate $x_{t+1}$, we write the Lagrangian:

$$L(x, \lambda, \tau) = \langle g, x \rangle + \frac{1}{\alpha_t} \sum_{j=1}^d x_j \log \frac{x_j}{x_{t,j}} - \langle \lambda, x \rangle + \tau (\langle I, x \rangle - 1).$$

Here, $\lambda \in \mathbb{R}^d$ is the multiplier for the constraint $x \geq 0$ and $\tau \in \mathbb{R}$ is the multiplier for the constraint $\langle I, x \rangle = 1$. Taking $\frac{\partial}{\partial x} L(x, \lambda, \tau) = 0$ gives

$$x_{t+1,j} = x_{t,j} \exp \left( -\alpha g_j + \lambda_j \alpha - \tau \alpha - 1 \right) > 0.$$

Hence the constraint $x \geq 0$ is inactive, which implies $\lambda = 0$. We choose $\tau$ to normalize $x$, giving

$$x_{t+1} = \left( \frac{x_{t,i} \exp \left( -\alpha_t g_{t,i} \right)}{\sum_{j=1}^d x_{t,j} \exp \left( -\alpha_t g_{t,j} \right)} \right)_{i=1,\ldots,d}$$  \hspace{1cm} (2)

$$\propto \left( x_{t,i} \exp \left( - \sum_{k=1}^t \alpha_k g_{k,i} \right) \right)_{i=1,\ldots,d}$$  \hspace{1cm} (3)

$$= \text{soft-argmin} \left\{ \sum_{k=1}^t \alpha_k g_{k,i}, \ i = 1, \ldots, d \right\}.$$  \hspace{1cm} (4)

**Remark 3.** In the context of the expert problem, $g_{k,i}$ is the loss of expert $i$ at time $k$. Hence $\sum_{k=1}^t g_{k,i}$ is the total loss of expert $i$ up to time $t$. Hence exponentiated gradient descent favors experts with low loss, but still assigns positive weight to every expert. This algorithm can thus be interpreted as a smoothed version of "follow the leader", where the weights are updated in an multiplicative fashion. (Variants of) exponentiated gradient descent is also known as **multiplicative weight update** (MWU), **follow-the-regularized-leader** (FTRL), **fictitious play** (FP), **Hedge algorithm**, and **entropic mirror descent**.
5 Analysis of Online Mirror Descent

We recall some definitions.

**Definition 2 (Strong convexity).** \( \psi \) is **strongly convex** with respect to \( \| \cdot \| \) if, for all \( y, x \):

\[
\psi(x) - \psi(y) - \langle g, x - y \rangle \geq \frac{1}{2} \| x - y \|^2,
\]

for all \( g \in \partial \psi(y) \).

This is equivalent to \( B_\psi(x, y) \geq \frac{1}{2} \| x - y \|^2 \) by definition of Bregman divergence.

**Example 1.** Let \( \psi(x) = \sum_j x_j \log x_j \) be negative entropy. Then by Pinsker’s inequality, we have

\[
B_\psi(x, y) = D_{KL}(x, y) \geq \frac{1}{2} \| x - y \|^2_1.
\]

In other words, the negative entropy is strongly convex with respect to the \( \ell_1 \) norm.

**Definition 3 (Dual norm).** The dual norm of \( \| \cdot \| \) is the norm \( \| \cdot \|_* \) defined by

\[
\| y \|_* = \sup_{x : \| x \| \leq 1} \langle x, y \rangle.
\]

**Example 2.** The dual norm of \( \| \cdot \|_2 \) is \( \| \cdot \|_2 \). The dual norm of \( \| \cdot \|_\infty \) is \( \| \cdot \|_1 \). The dual norm of \( \| \cdot \|_{\text{nuc}} \) (nuclear norm) is \( \| \cdot \|_{\text{op}} \) (operator norm).

**Theorem 1.** Suppose that \( \psi \) is strongly convex with respect to \( \| \cdot \| \) with dual norm \( \| \cdot \|_* \). Then online mirror descent with step size \( \alpha_t \equiv \alpha \) satisfies

\[
\sum_{t=1}^T [f_t(x_t) - f_t(x^*)] \leq \frac{1}{\alpha} B_\psi(x^*, x_1) + \frac{\alpha}{2} \sum_{t=1}^T \| g_t \|_*^2.
\]

**Proof.** Recall that \( x_{t+1} = \arg\min_{x \in X} \{ \langle g_t, x \rangle + \frac{1}{\alpha} B_\psi(x, x_t) \} \). By the optimality condition for constrained optimization (negative gradient lies in the normal cone), we have

\[
0 \leq \left. \left( g_t + \frac{1}{\alpha} \frac{\partial}{\partial x} B_\psi(x, x_t) \right) \right|_{x = x_{t+1}}, \quad x^* - x_{t+1}
\]

\[
= \left( g_t + \frac{1}{\alpha} (\nabla \psi(x_{t+1}) - \nabla \psi(x_t)) , x^* - x_{t+1} \right).
\]

Therefore, we have

\[
f_t(x_t) - f_t(x^*) \leq \langle g_t, x_t - x^* \rangle \quad \text{convexity of } f_t
\]

\[
= \langle g_t, x_{t+1} - x^* \rangle + \langle g_t, x_t - x_{t+1} \rangle
\]

\[
\leq \frac{1}{\alpha} \langle \nabla \psi(x_t) - \nabla \psi(x_t), x^* - x_{t+1} \rangle + \langle g_t, x_t - x_{t+1} \rangle \quad \text{last display equation}
\]

\[
= \frac{1}{\alpha} \left[ B_\psi(x^*, x_t) - B_\psi(x^*, x_{t+1}) - B_\psi(x_{t+1}, x_t) \right] + \langle g_t, x_t - x_{t+1} \rangle,
\]
where the last step follows from direct calculation using definition and is sometimes known as the “three-point identity” (HW2 Q3.3). Summing over \( t = 1, \ldots, T \), the sum telescopes, and we get

\[
\sum_{t=1}^{T} (f_t(x_t) - f_t(x^*)) \leq \frac{1}{\alpha} \left[ B_{\psi}(x^*, x_1) - B_{\psi}(x^*, x_{T+1}) \right] + \frac{1}{\alpha} \sum_{t=1}^{T} \left[ -B_{\psi}(x_{t+1}, x_t) + \langle g_t, x_t - x_{t+1} \rangle \right]
\]

To control the last RHS term, we observe that

\[
\langle g_t, x_t - x_{t+1} \rangle \leq \|g_t\| \|x_t - x_{t+1}\| \quad \text{definition of dual norm}
\]

\[
\leq \frac{\alpha}{2} \|g_t\|^2 + \frac{1}{2\alpha} \|x_t - x_{t+1}\|^2 \quad \text{strong convexity of } \psi.
\]

Combining pieces, we obtain the desired regret bound

\[
\sum_{t=1}^{T} (f_t(x_t) - f_t(x^*)) \leq \frac{1}{\alpha} B_{\psi}(x^*, x_1) + \frac{\alpha}{2} \sum_{t=1}^{T} \|g_t\|^2.
\]

\[\square\]

6 Applications

6.1 Online (sub)-gradient descent

Let \( \psi(x) = \frac{1}{2} \|x\|^2 \). Then \( \psi \) is strong convex with respect to \( \|\cdot\|_2 \), and the dual norm is \( \|\cdot\|_2 \). Suppose each \( f_t \) is \( L \)-Lipschitz, which implies \( \|g_t\|_2 \leq M \). Then the regret bound is

\[
\sum_{t=1}^{T} (f_t(x_t) - f_t(x^*)) \leq \frac{1}{2\alpha} \|x^* - x_1\|^2 + \frac{\alpha}{2} T \cdot M^2.
\]

Choosing \( \alpha = \frac{\|x^* - x_1\|^2}{M \sqrt{T}} \) to minimize the RHS gives

\[
\frac{1}{T} \sum_{t=1}^{T} (f_t(x_t) - f_t(x^*)) \leq \frac{\|x^* - x_1\|^2 M}{\sqrt{T}}.
\]

Remark 4. The above bound implies an \( O(\frac{1}{\sqrt{T}}) \) convergence rate for the offline setting where all \( f_t \equiv f \). In particular, letting \( \bar{x} = \frac{1}{T} \sum_{t=1}^{T} x_t \), we have

\[
f(\bar{x}) - f(x^*) \leq \frac{1}{T} \sum_{t=1}^{T} [f(x_t) - f(x^*)] \leq \frac{\|x^* - x_1\|^2 M}{\sqrt{T}},
\]

where the first step above is by Jensen’s inequality. This recovers the result from Lecture 17 on subgradient descent.
6.2 Exponentiated gradient descent

Let \( X = \Delta_d \), and \( \psi(x) = \sum_j x_j \log x_j \) be the negative entropy. Then \( \psi \) is strongly convex with respect to \( \| \cdot \|_1 \), with dual norm \( \| \cdot \|_\infty \). Then the regret bound is

\[
\sum_{t=1}^T (f_t(x_t) - f_t(x^*)) \leq \frac{1}{\alpha} D_{KL}(x^*, x_1) + \frac{\alpha}{2} \sum_{t=1}^T \| g_t \|_\infty^2.
\]

If in addition we take the initial iterate \( x_1 = \left( \frac{1}{d}, \ldots, \frac{1}{d} \right) \) to be the uniform distribution, then one can verify that \( D_{KL}(x^*, x_1) \leq \log d \). Also, set \( \alpha = \sqrt{\frac{\log d}{2T \max_t \| g_t \|_\infty}} \). Then the average regret is

\[
\frac{1}{T} \sum_{t=1}^T (f_t(x_t) - f_t(x^*)) \leq \sqrt{\frac{\log d \cdot \max_t \| g_t \|_\infty^2}{T}}.
\]

**Remark 5.** Compared to online gradient descent, the dependence on the gradients \( g_t \) is \( \max_t \| g_t \|_\infty \) instead of \( \max_t \| g_t \|_2 \). Thus exponentiated gradient descent can do better than gradient descent when the gradients \( g_t \) are small in magnitude and not sparse.

6.3 Expert problem

Recall that \( l_{tj} \) is the loss of expert \( j \) at time \( t \), and that \( g_t = l_t \in \{0, 1\}^d \). Thus \( \| g_t \|_\infty \leq 1 \). Plugging this into the bound for exponentiated gradient descent gives

\[
\frac{1}{T} \sum_{t=1}^T (f_t(x_t) - f_t(x^*)) \leq \sqrt{\frac{\log d}{T}}.
\]

**Remark 6.** This regret bound is optimal for the expert problem. In comparison, gradient descent would get \( \sqrt{d} \) regret, which has an exponentially larger dependence on the dimension \( d \).

7 Extensions

1. We chose our step size \( \alpha \) to be proportional to \( \frac{1}{\sqrt{T}} \). This requires the time horizon to be known to the algorithm. If \( T \) is not known, one can use a varying step size \( \alpha_t = \frac{1}{\sqrt{t}} \) and prove essentially the same guarantees (under a slightly stronger boundedness assumption; see Duchi’s notes.)

2. **Improve bounds.** If more is known about the loss function \( f_t \), then better regret bounds (in the online setting) and convergence rates (in the offline setting) can be obtained.

   - \( f_t \) is smooth (gradient is Lipschitz): We have an improvement \( \sqrt{T} \to O(1) \) in regret, which translates to an improvement \( \frac{1}{\sqrt{T}} \to \frac{1}{T} \) in rate.

   - \( f_t \) is strongly convex: We have an improvement \( \sqrt{T} \to \log T \) in regret, and hence \( \frac{1}{\sqrt{T}} \to \frac{\log T}{T} \) in rate.

See Xinhua Zhang’s notes for details.
3. So far, we assumed that we observe the losses of all the experts/arms, even those we did not choose/pull. This is the full information setting. Next week, we will look at the “bandit information” setting, where we only observe the loss of the expert/arm that we choose/pull, that is, we only see one entry of $\nabla f_t = g_t = l_t$. 