CS 294-128: Algorithms and Uncertainty	Lecture 14	Date: October 17, 2016
Instructor: Nikhil Bansal		Scribe: Antares Chen

1 Introduction

In this lecture, we review results regarding follow the regularized leader (FTRL). We then begin to discuss a new online convex optimization algorithm known as mirror descent. First, we construct the intuition behind the algorithm by introducing Bregman divergence. We then discuss the mechanics of the mirror descent algorithm, show remarkable equivalence with FTRL, and provide an example application. Finally, we relate online mirror decent to Fenchel Duality and provide some intuition behind using Bregman divergence as a distance metric.

2 Review

2.1 Setting

For the past few lectures, we have discussed online convex optimization (OCO). The problem specifications are as follows. We are given some decision domain modeled as a convex set $\mathcal{K}$ in Euclidean space. At each time step t, the player is hit with a convex cost function $f_t : \mathcal{K} \to \mathbb{R}$. The player then chooses x_t such that the regret is minimized.

$$regret = \sum_t f_t(\mathbf{x}_t) - \min_{\mathbf{y} \in \mathcal{K}} \sum_t f_t(\mathbf{y})$$

For the remainder of these notes, we denote $\nabla_i = f_i(\mathbf{x}_i)$ and assume all cost functions are linear. Our regret analysis will also depend on the notion of *diameter* which we now define.

Definition 1 *The diameter with respect to R is given by*

$$D_R = \sqrt{\max_{\mathbf{x},\mathbf{y}} \{R(\mathbf{x}) - R(\mathbf{y})\}}$$

2.2 Follow the regularized leader

Previously, we discussed an online convex optimization algorithm known as follow the regularized leader (FTRL) which was introduced in [5][6]. The analysis of online mirror descent will rely heavily on this topic and so we review the algorithm here. At the t-th time step, the next value $\mathbf{x}_{t+1}$ is chosen based on this update rule.

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x} \in \mathcal{K}} \left\{ \eta(\nabla_1 + \cdots + \nabla_t) \cdot \mathbf{x} + R(\mathbf{x}) \right\}$$

Here, $R(\mathbf{x})$ is a regularization function that is often chosen to be α-strongly convex with respect to some norm $|| \cdot ||$. Analysis based on the regime *be the leader* (BTL) [2] yielded the following regret bound.

Theorem 2 *Let $||\cdot||_*$ denotes the dual norm with respect to $||\cdot||$. If $R(\mathbf{x})$ is α-strongly convex with respect to $||\cdot||$. Then the regret for FTRL is bounded as follows.*

$$regret \leq \sum_t \frac{2\eta}{\alpha}||\nabla_t||_*^2 + \frac{R(\mathbf{y}) - R(\mathbf{x})}{\eta}$$

3 Online Mirror Descent

We now introduce *online mirror descent* (OMD) which is an online variant of Nemirovski and Yudin's mirror descent algorithm [4]. First discussed by [7], OMD is very similar online gradient descent as the algorithm computes the current decision iteratively based on a gradient update rule and the previous decision. However, the power behind OMD lies in the update being carried out in a "dual" space, defined by our choice of regularizer. This follows from considering ∇R as a mapping from $\mathbb{R}^n$ onto itself.

When carrying out the update in this space, we take advantage of a rich geometry defined only in the dual. Indeed, this has lead to discoveries that show many algorithms to be *special cases* of online mirror descent [3][9]. More recently, it has been discovered that not only does online mirror descent apply to a *general* class of online convex optimization problems, but that they do so with optimal regret bounds [8].

3.1 The algorithm

Online mirror descent will rely on *Bregman divergence*.

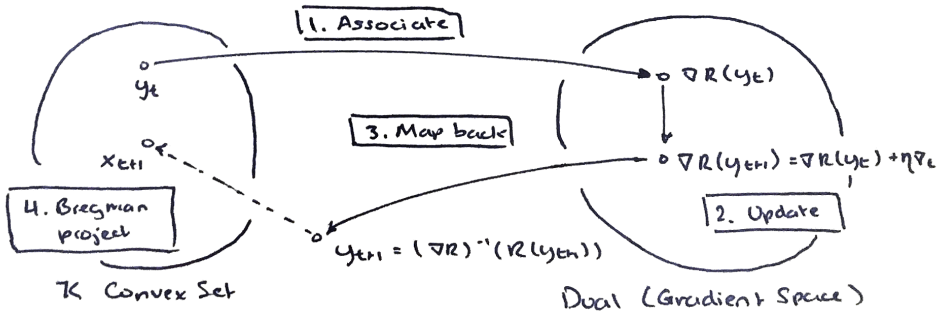
Definition 3 *Denote $B_R(\mathbf{x}||\mathbf{y})$ as the Bregman divergence between $\mathbf{x}$ and $\mathbf{y}$ with respect to the function R. This is given as*

$$B_R(\mathbf{x}||\mathbf{y}) = R(\mathbf{x}) - R(\mathbf{y}) - \nabla R(\mathbf{y}) \cdot (\mathbf{x} - \mathbf{y})$$

We immediately have the notion of a *Bregman projection* of $\mathbf{y}$ onto a convex set $\mathcal{K}$.

$$\operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} B_R(\mathbf{x}||\mathbf{y})$$

We are now ready to discuss online mirror descent. The algorithm takes in as input the learning rate $\eta > 0$ and regularization function $R(\mathbf{x})$. Graphically, the algorithm runs as follows.



The pseudocode is provided below.

Algorithm 1 Online mirror descent

1: Initialize $\mathbf{y}_1$ to be such that $\nabla R(\mathbf{y}_1) = 0$ and $\mathbf{x}_1 = \text{argmin}_{\mathbf{x}\in\mathcal{K}} B_R(\mathbf{x}||\mathbf{y}_1)$
2: **for** $t = 1 \to T$ **do**
3: Play $\mathbf{x}_t$ and receive cost function f_t
4: Update y_t according to the following rule

$$\nabla R(\mathbf{y}_{t+1}) = \nabla R(\mathbf{y}_t) - \eta \nabla_t$$

5: Bregman project back to $\mathcal{K}$

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} B_R(\mathbf{x}||\mathbf{y}_{t+1})$$

6: **end for**

In terms of implementation, $\mathbf{y}_{t+1}$ may be recovered by applying the inverse gradient mapping $(\nabla R)^{-1}$. In general, if R is α-strongly convex, then ∇R must be a bijective mapping.

3.2 Regret analysis

Hazan and Kale [1] provided an extraordinary result equating FTRL with OMD. This theorem, which we now prove below, will in the future allow us to bootstrap theorem 2 and provide regret bounds for online mirror descent.

Theorem 4 *Given that R is α-strongly convex, the lazy OMD and FTRL algorithms produce equivalent predictions.*

$$\operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} B_R(\mathbf{x}||\mathbf{y}_{t+1}) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \Big(\eta \sum_{s=1}^{t} \nabla_s \cdot \mathbf{x} + R(\mathbf{x})\Big)$$

Proof: Observe that in lazy OMD, $\mathbf{y}_{t+1}$ is updated with respect to the following constraint $\nabla R(\mathbf{y}_{t+1}) = \nabla R(\mathbf{y}_t) - \eta\nabla_t$. This gives us the following.

$$\begin{aligned}
\mathbf{y}_{t+1} &= (\nabla R)^{-1}\big(\nabla R(\mathbf{y}_t) - \eta\nabla_t\big) \\
&= (\nabla R)^{-1}\big(\nabla R(\mathbf{y}_{t-1}) - \eta\nabla_{t-1} - \eta\nabla_t\big) \\
&= (\nabla R)^{-1}\Big(-\sum_{s=1}^{t} \eta\nabla_s\Big)
\end{aligned}$$

Consider the case where $\mathbf{y}_{t+1} \in \mathcal{K}$ implying that the projection is itself. In the OMD regime, we have that $\mathbf{x}_{t+1} = \mathbf{y}_{t+1}$. Denote the FTRL update as $\Phi_t = \sum_{s=1}^{t} \eta\nabla_s \cdot \mathbf{x} + R(\mathbf{x})$. Taking the gradient gives us the following.

$$\nabla\Phi_t = \sum_{s=1}^{t} \eta\nabla_s + \nabla R(\mathbf{x})$$

However, in FTRL after this quantity is minimized, we must have $\nabla \Phi_t = 0$.

$$\nabla R(\mathbf{x}) = -\sum_{s=1}^{t} \eta \nabla_s \qquad \mathbf{x} = (\nabla R)^{-1}\Big(-\sum_{s=1}^{t} \eta \nabla_s\Big)$$

Which is exactly $\mathbf{y}_{t+1}$.

Now if $\mathbf{y}_{t+1} \notin \mathcal{K}$, we must then Bregman project back to $\mathcal{K}$. This is given by definition, but since we minimize with respect to $\mathbf{x}$, terms independent of this variable can be eliminated giving us the following.

$$\begin{aligned}
\operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} B_R(\mathbf{x}||\mathbf{y}_{t+1}) &= \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \big\{R(\mathbf{x}) - R(\mathbf{y}_{t+1}) - \nabla R(\mathbf{y}_{t+1}) \cdot (\mathbf{x} - \mathbf{y}_{t+1})\big\} \\
&= \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \big\{R(\mathbf{x}) - \nabla R(\mathbf{y}_{t+1}) \cdot \mathbf{x}\big\} \\
&= \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \big\{R(\mathbf{x}) + \sum_{s=1}^{t} \eta \nabla_s \cdot \mathbf{x}\big\}
\end{aligned}$$

In all cases, the updates for OMD and FTRL are equivalent. Thus the theorem holds. □

4 Experts From Online Mirror Descent

As stated previously, many algorithms occur as special cases of online mirror descent. We now showcase the results of [3].

Recall the setup for experts. At time t a probability distribution p_t is maintained on k experts and a loss vector ℓ_t is revealed. Our goal is to maximize the probability of picking the expert i who incurs minimal loss over T time steps.

4.1 Exponentiated gradient algorithm

Let $\mathbf{x}(i)$ be the i-th component of $\mathbf{x}$ and our regularization function be the negative entropy function $R(\mathbf{x}) = \sum_i \mathbf{x}(i) \log \mathbf{x}(i)$. We then have that $\nabla R(x) = \sum_i (\log \mathbf{x}(i) + 1)$. From the OMD algorithm, our update rule for $\mathbf{y}_{t+1}$ is then the following.

$$\begin{aligned}
\nabla R(\mathbf{y}_{t+1}) &= \nabla R(\mathbf{y}_t) - \eta \nabla_t \\
\sum_i (\log \mathbf{y}_{t+1}(i) + 1) &= \sum_i (\log \mathbf{y}_t(i) + 1) - \eta \nabla_t \\
\sum_i \log \mathbf{y}_{t+1}(i) &= \sum_i \log \mathbf{y}_t(i) - \eta \nabla_t \\
\mathbf{y}_{t+1}(i) &= \mathbf{y}_t(i) e^{-\eta \nabla_t}
\end{aligned}$$

Recall that in the expert setting, our convex set $\mathcal{K}$ is simply the n-dimensional simplex defined as $\Delta_n = \{\mathbf{x} \in \mathbb{R}^n : \sum_i \mathbf{x}(i) = 1\}$. We make two critical observations.

- By theorem 6, the Bregman divergence with respect to the negative entropy function becomes relative entropy. This is also known as Kullback-Liebler (KL) divergence.

- By theorem 7, the Bregman projection with respect to the negative entropy function becomes scaling by the ℓ_1-norm.

We have fully defined a special case of the OMD update regime called the *exponentiated gradient algorithm*.

Algorithm 2 Exponentiated gradient

1: Initialize $\mathbf{y}_1 = \mathbf{1}$ and $\mathbf{x}_1 = \frac{\mathbf{y}_1}{||\mathbf{y}_1||_1}$
2: **for** $t = 1 \to T$ **do**
3: Play $\mathbf{x}_t$ and receive cost function f_t
4: Update $\mathbf{y}_t$ according to the following rule

$$\mathbf{y}_{t+1}(i) = \mathbf{y}_t(i)e^{-\eta\nabla_t(i)}$$

5: Bregman project back to $\mathcal{K}$

$$\mathbf{x}_{t+1} = \frac{\mathbf{y}_{t+1}}{||\mathbf{y}_{t+1}||_1}$$

6: **end for**

Previously, we provided a multiplicative weight update method for expert learning and proved regret bounds using a potential function argument. However, here the algorithm directly falls out of OMD as a special case!

4.2 Regret analysis

We have demonstrated that OMD is equivalent to FTRL and so we may bootstrap theorem 2 to bound the regret of exponentiated gradient.

Theorem 5 *Suppose all expert costs are 0-1 bounded: $\ell_t(i) \in [0, 1]$. Then the regret for the exponentiated gradient algorithm is given by*

$$regret \leq O\big(\sqrt{T \log n}\big)$$

Proof: First, substitute $R(\mathbf{y}) - R(\mathbf{x})$ with diameter. By theorem 2, we have the following.

$$regret \leq \sum_t \frac{2\eta}{\alpha}||\nabla_t||_*^2 + \frac{D_R^2}{\eta}$$

Differentiate with respect to η and minimize the above expression.

$$\eta = \sqrt{\frac{\alpha D_R^2}{2\sum_t ||\nabla_t||_*^2}} \qquad \Rightarrow \qquad regret \leq 2D_R\sqrt{\sum_t \frac{2}{\alpha}||\nabla_t||_*^2}$$

Observe that if all expert costs are in the range $[0, 1]$, then the cost gradient must be bounded in the following manner.

$$||\nabla_t||_* = ||\nabla_t||_\infty \leq 1$$

By Pinsker's inequality (theorem 8), the negative entropy function is strongly convex with respect to the ℓ_1-norm. However, the dual of the ℓ_1-norm is the ℓ_∞-norm, which follows from generalize Cauchy-Schwartz.

Additionally, the negative entropy function is α-strongly convex where $\alpha = \frac{1}{2\ln 2}$. Using Jensen's inequality, one may show that $D_R \leq \sqrt{\log n}$ on the simplex Δ_n. Our regret is now the following.

$$regret \leq 2D_R\sqrt{\sum_t \frac{2}{\alpha}||\nabla_t||_*^2} = 2\sqrt{\sum_t \frac{2\log n}{2\ln 2}} = 2\sqrt{\frac{T\log n}{\ln 2}} = O\big(\sqrt{T\log n}\big)$$

Thus completes our analysis. □

References

[1] E. Hazan and S. Kale. Extracting certainty from uncertainty: Regret bounded by variation in costs. In *The 21st Annual Conference on Learning Theory (COLT)*, pages 5768, 2008.

[2] A. Kalai and S. Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291307, 2005.

[3] J. Kivinen and M. Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *Information and Computation*, 132(1):164, 1997.

[4] A. Nemirovski and D. Yudin. On cesaros convergence of the gradient descent method for finding saddle points of convex-concave functions. *Doklady Akademii Nauk SSSR*, 239(4), 1978.

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

[6] S. Shalev-Shwartz and Y. Singer. A primal-dual perspective of online learning algorithms. *Machine Learning*, 69(2-3):115142, 2007.

[7] S. Shalev-Shwartz and Y. Singer. Convex repeated games and fenchel duality. *Advances in Neural Information Processing Systems*, 19:1265, 2007.

[8] N. Srebro, K. Sridharan, and A. Tewari. On the universality of online mirror descent. *Advances in Neural Information Processing Systems* pages. 2645-2653, 2001.

[9] M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. *ICML*, 2003.

A The Negative Entropy Function

In this section we provide calculations that show properties relevant to using negative entropy as the regularizer.

Theorem 6 *Let $R(\mathbf{x}) = \sum_i \mathbf{x}(i) \log \mathbf{x}(i)$. We have the following.*

$$B_R(\mathbf{x}||\mathbf{y}) = \sum_i \mathbf{x}(i) \log \Big(\frac{\mathbf{x}(i)}{\mathbf{y}(i)}\Big) - \sum_i \mathbf{x}(i) + \sum_i \mathbf{y}(i)$$

Proof: Calculations follow from definition. Note that $\nabla R(x) = \sum_i (\log \mathbf{x}(i) + 1)$.

$$\begin{aligned}
B_R(\mathbf{x}||\mathbf{y}) &= R(\mathbf{x}) - R(\mathbf{y}) - \nabla R(\mathbf{y}) \cdot (\mathbf{x} - \mathbf{y}) \\
&= \sum_i \mathbf{x}(i) \log \mathbf{x}(i) - \sum_i \mathbf{y}(i) \log \mathbf{y}(i) - \sum_i \big(\log \mathbf{y}(i) + 1\big)\big(\mathbf{x}(i) - \mathbf{y}(i)\big) \\
&= \sum_i \mathbf{x}(i) \log \mathbf{x}(i) - \sum_i \mathbf{x}(i) \log \mathbf{y}(i) - \sum_i \mathbf{x}(i) + \sum_i \mathbf{y}(i) \\
&= \sum_i \mathbf{x}(i) \log \Big(\frac{\mathbf{x}(i)}{\mathbf{y}(i)}\Big) - \sum_i \mathbf{x}(i) + \sum_i \mathbf{y}(i)
\end{aligned}$$

The theorem holds. □

Noticeably, we have that the Bregman divergence of negative entropy is simply KL-divergence. Given this formulation we prove the following.

Theorem 7 *Let $R(\mathbf{x}) = \sum_i \mathbf{x}(i) \log \mathbf{x}(i)$. Then $B_R(\mathbf{x}||\mathbf{y})$ subject to $\mathbf{x} \in \Delta_n$ is minimized at the following point.*

$$\mathbf{x} = \left\langle \frac{\mathbf{y}(i)}{||\mathbf{y}||_1} \right\rangle$$

Proof: We wish to minimize the following expression with respect to $\mathbf{x}$ subject to $\sum_i \mathbf{x}(i) = 1$.

$$\mathbf{x} = \operatorname*{argmin}_{\mathbf{x} \in \Delta_n} \Big\{ \sum_i \mathbf{x}(i) \log \Big(\frac{\mathbf{x}(i)}{\mathbf{y}(i)}\Big) - \sum_i \mathbf{x}(i) + \sum_i \mathbf{y}(i) \Big\}$$

This is easily done using Lagrange multipliers. Let F be defined as follows.

$$F(\mathbf{x}, \lambda) = \sum_i \mathbf{x}(i) \log \Big(\frac{\mathbf{x}(i)}{\mathbf{y}(i)}\Big) - 1 + \sum_i \mathbf{y}(i) - \lambda \Big(\sum_i \mathbf{x}(i) - 1 \Big)$$

One can show that $\partial F / \partial \mathbf{x}(i) = 0$ at the following values.

$$\mathbf{x}(i) = \mathbf{y}(i) e^{\lambda - 1} \qquad\qquad \lambda = \frac{1}{\ln \sum_i \mathbf{y}(i)} + 1$$

Substituting in gives us the theorem. □

This gives us the interpretation that the Bregman projection with respect to negative entropy on the n-dimensional simplex becomes scaling by the ℓ_1-norm.

B Pinsker's inequality

In this section, we prove Pinsker's inequality which gives us the fact that negative entropy is α-strongly convex with respect to the ℓ_1-norm given $\alpha = \frac{1}{2\ln 2}$.

Theorem 8 *Let P and Q be two distributions defined on the sample space Ω. Then the following holds.*

$$D_{KL}(P||Q) \geq \frac{1}{2\ln 2} \cdot ||P - Q||_1^2$$

Proof: We first show the theorem holds for the case where P and Q are Bernoulli distributions. Let $p, q \in [0, 1]$ and P, Q given by the following.

$$P = \begin{cases} 1 & \text{w.p. } p \\ 0 & \text{w.p. } 1-p \end{cases} \qquad Q = \begin{cases} 1 & \text{w.p. } q \\ 0 & \text{w.p. } 1-q \end{cases}$$

Without loss of generality, let $p \geq q$ and define f to be the following.

$$\begin{aligned} f(p,q) &= D_{KL}(P||Q) - \frac{1}{2\ln 2}||P - Q||_1^2 \\ &= p\log\frac{p}{q} + (1-p)\log\frac{1-p}{1-q} - \frac{4(p-q)^2}{2\ln 2} \end{aligned}$$

And observe that we have $f(p, q) = 0$ when $p = q$ and $f(p, q) \geq 0$ when $q \leq p$. Furthermore, the following holds.

$$\frac{\partial f}{\partial q} = -\frac{p-q}{\ln 2}\Big(\frac{1}{q(1-q)} - 4\Big)$$

We conclude that $D_{KL}(P||Q) \geq \frac{1}{2\ln 2}||P - Q||_1^2$. Now consider the case where P and Q are distributed arbitrarily on Ω. Let $A \subseteq \Omega$ be such that $A = \{x : P(x) \geq Q(x)\}$ and define the following random variables.

$$P_A = \begin{cases} 1 & \text{w.p. } \sum_{x\in A} P(x) \\ 0 & \text{w.p. } \sum_{x\notin A} P(x) \end{cases} \qquad Q_A = \begin{cases} 1 & \text{w.p. } \sum_{x\in A} Q(x) \\ 0 & \text{w.p. } \sum_{x\notin A} Q(x) \end{cases}$$

We then have the following.

$$\begin{aligned} ||P - Q||_1 &= \sum_{x\in\Omega} \big|P(x) - Q(x)\big| \\ &= \sum_{x\in A} \big(P(x) - Q(x)\big) + \sum_{x\notin A} \big(Q(x) - P(x)\big) \\ &= \left|\sum_{x\in A} P(x) - \sum_{x\notin A} Q(x)\right| + \left|\Big(1 - \sum_{x\in A} P(x)\Big) - \Big(1 - \sum_{x\notin A} Q(x)\Big)\right| \\ &= ||P_A - Q_A||_1 \end{aligned}$$

Now define the random variable Z to be $Z(x) = 1$ if $x \in A$ else $Z(x) = 0$. It follows that $D_{KL}(P||Q) = D_{KL}(P(Z))||Q((Z)) + D_{KL}(P||Q|Z)$. However, $D_{KL}(P(Z))||Q((Z)) = D_{KL}(P_A||Q_A)$ and $D_{KL}(P||Q|Z) \geq 0$, we must have the following.

$$D_{KL}(P||Q) \geq D_{KL}(P_A||Q_A) \geq \frac{1}{2\ln 2}||P_A - Q_A||_1^2 = \frac{1}{2\ln 2}||P - Q||_1^2$$

Thus we complete the proof. □