IIIS 2018 Spring: ATCS - Selected Topics in Optimization
Lecture date: March 19, 2018
Instructor: Jian Li **Scribe: Niu Hui**

1 Online Mirror Descent

In the last lecture we have covered the knowledge of dual norm and Fenchel conjugate. In this post we are interested in minimizing a convex function f over a compact convex set $X \subseteq \mathbb{R}^n$ with some assumption of f.

1.1 Mirror Map

Definition 1 (Mirror Map) *With domain D, a mirror map is a function $F : D \to \mathbb{R}^+$ satisfying:*

① *Strongly convex (w.r.t. $\| \cdot \|$)*
α-strongly convex:

$$f(y) \geq f(x) + \nabla f(x)^T \cdot (y - x) + \frac{\alpha}{2} \| x - y \|^2$$

② $\nabla F(D) =: \{\nabla F(x | x \in D)\} = \mathbb{R}^n$

③ $\lim_{x \to \partial D} \| \nabla F(D) \| \to \infty$

Note that the second condition requires the gradient space is the whole space. The third condition means the gradients of the points near the margin of D are close to ∞.

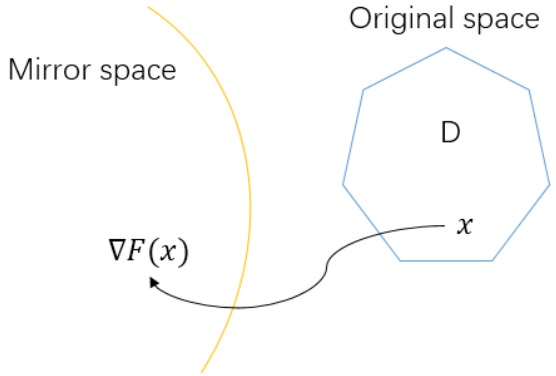

Figure 1: Map the original space to the mirror gradient space

Example 1.1 *For $\| \cdot \|_2$, if $F(x) = \frac{1}{2}x^2$, we project $x \to \nabla F(x) = x$, the Bregman divergence is*

$$D_F(x, y) = F(x) - F(y) - \langle \nabla F(y), x - y \rangle = \frac{1}{2} \| x - y \|^2$$

.

Example 1.2 (neg-entropy function) *If $F(x) = \sum_i x_i \log x_i$ (neg-entropy function), $F(x)$ is 1-strongly convex w.r.t. $\|\cdot\|_1$. We have*

$$\nabla F(x) = \begin{pmatrix} \log x_1 + 1 \\ ..... \\ \log x_n + 1 \end{pmatrix}$$

and the Bregman divergence of F is

$$D_F(x, y) = \sum_{i=1}^{n} x_i \log \frac{x_i}{y_i} - \sum_{i=1}^{n} (x_i - y_i).$$

The first part $\sum_{i=1}^{n} x_i \log \frac{x_i}{y_i}$ is KL-divergence. Note that if x and y are from the same distribution, the KL-divergence are shrink to 0.

Check by yourself: The calculation of the Bregman divergence in example 1.2 is as following

$$\begin{aligned} D_F(x, y) &= F(x) - F(y) - \langle \nabla F(y), x - y \rangle \\ &= \sum_i x_i \log x_i - \sum_i y_i \log y_i - \sum_i (x_i - y_i)(\log y_i + 1) \\ &= \sum_{i=1}^{n} x_i \log \frac{x_i}{y_i} + \sum_i (x_i - y_i). \end{aligned}$$

To build and illustrate the algorithm of Online Mirror Descent (OMD), now we give some lemmas at first.

Recall: The Fenchel dual/conjugate of F is

$$F^\star(\theta) = \sup_{x \in D} (\langle x, \theta \rangle - F(x)).$$

The Bregman divergence of F is

$$D_F(x, y) = F(x) - F(y) - \langle \nabla F(y), x - y \rangle$$

.

Lemma 2 *If F is a mirror map (Legendre function), and $F^\star$ is the Fenchel conjugate of F.*

① $F^{\star\star} = F$

② $\nabla(F^\star) = (\nabla F)^{-1}$

③ $D_F(x, y) = D_{F^\star}\Big(\nabla F(x), \nabla F(y)\Big)$

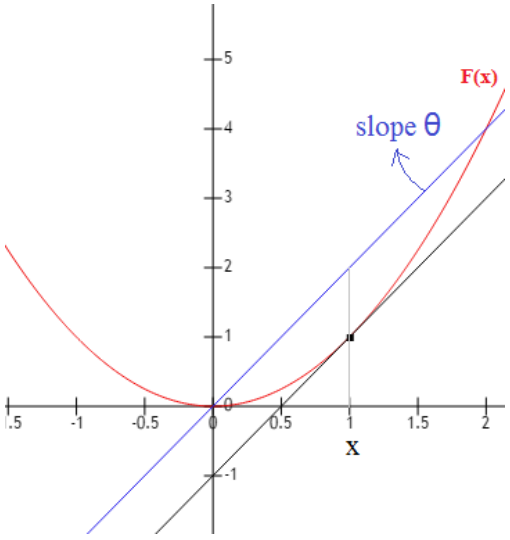

Figure 2: The Fenchel Conjugate

Proof of Lemma 2-②. In homework 2 we will prove

$$\nabla F^\star(\theta) = \arg\max_x(\langle x, \theta\rangle) =: x_0.$$

See the Figure 2, we know

$$\nabla F(x_0) = \theta.$$

So we have

$$(\nabla F)^{-1}(\theta) = x_0.$$

Then we proved the lemma 2-②. □

Lemma 3 (Generalized Triangle Equality) *Let D_f denote the Bregman divergence of f.*

$$\Big(\nabla f(z) - \nabla f(x)\Big)^T \cdot (x - y) = D_f(x, y) + D_f(z, x) - D_f(z, y)$$

Proof: The right part is

$$\begin{aligned} D_f(x,y) + D_f(z,x) - D_f(z,y) =& + f(x) - f(y) - \nabla f(x)^T(x-y) \\ &+ f(z) - f(x) - \nabla f(z)^T(z-x) \\ &- f(z) + f(y) + \nabla f(z)^T(z-y) \\ =& \Big(\nabla f(z) - \nabla f(x)\Big)^T \cdot (x-y) \end{aligned}$$

□

1.2 Online Mirror Descent Algorithm (OMD)

Now we introduce the Online Mirror Descent (OMD) algorithm, see algorithm 1 for details.

To illustrate OMD (see Figure), we can imagine the algorithm first project x_t to the point $\nabla F(x_t)$ in mirror gradient space. Then move a step in the direction of $-\eta\nabla f_t(x_t)$. Then we want

Algorithm 1: Online Mirror Descent (OMD)

for $t = 1, ..., T$ **do**
- ① play x_t, nature plays f_t(incur a loss func $f_t(x_t)$);
- ② $W_{t+1} \leftarrow \nabla F^\star\big(\nabla F(x_t) - \eta \nabla f_t(x_t)\big)$;
- ③ $x_{t+1} \leftarrow \arg\min_{y \in D} D_F(y, W_{t+1})$;

to project this point back to the original space. Hence, we use the inverse function $(\nabla F)^{-1} = \nabla F^\star$. However, the inverse function may not drop in the original space. Thus we need to get it back. That is the last step of the algorithm.

Example 1.3 *(Projected GD) If we use $F(x) = \frac{1}{2}x^2$ in projected gradient descent, it is the Mirror Descent algorithm.*

Recall: Last time we discussed the universal portfolio. For the simplex, n-experts problem, the regret bound of project GD is

$$reg_T \leq D \cdot \max \parallel \nabla f \parallel \sqrt{T} \leq \sqrt{nT}.$$

However the regret bound of Multiplicative Weight (MW) is

$$reg_T \leq \sqrt{T \ln n}.$$

Now we show when using different mirror maps F, the regret bound of projected GD may be as good as the MW algorithm.

Theorem 4 *Let*

$$\eta = \sqrt{\frac{2 \sup_{x \in D} F(x)}{T \cdot B^2}},$$
$$B = \sup_{t,x} \parallel \nabla f_t(x) \parallel_\star .$$

The regret bound of OMD is

$$Reg_T \leq \sqrt{\frac{2B^2 \sup_{x \in D} F(x)}{T}}.$$

Proof: Without loss of generality, $f_t(x) = \langle \nabla_t, x_t \rangle$ is a linear function.

$$\begin{aligned} LHS_t &= \langle \nabla_t, x_t \rangle - \langle \nabla_t, x^\star \rangle \\ &= \langle \nabla_t, x_t - W_{t+1} + W_{t+1} - x^\star \rangle \\ &= \langle \nabla_t, x_t - W_{t+1} \rangle + \frac{1}{\eta} \big\langle (\nabla F(x_t) - \nabla F(W_{T+1})), W_{t+1} - x^\star \big\rangle \end{aligned}$$

To understand the last line of the formula, we apply the algorithm 1-② and obtain

$$\begin{aligned} ALGO1-②: & W_{t+1} \leftarrow \nabla(F^\star)(\nabla F(x_t) - \eta\nabla_t) \\ & \Downarrow \\ & \nabla F(W_{t+1}) \leftarrow \nabla F(x+t) - \eta\nabla_t \\ & \Downarrow \\ & \nabla_t = \frac{1}{\eta}\big(\nabla F(x_t) - \nabla F(W_{T+1})\big) \end{aligned}$$

Applying the lemma 3 (Generalized triangle equation) and the fact $\langle a, b\rangle \leq \| a \|_\star \cdot \| b \| \leq \frac{\eta}{2} \| a \|_\star^2 + \frac{1}{2\eta} \| b \|^2$, we obtain

$$\begin{aligned} LHS_t &= \langle \nabla_t, x_t - W_{t+1}\rangle + \frac{1}{\eta}\big(D_F(x^\star, x_t) - D_F(x^\star, W_{t+1}) - D_F(x_t, , W_{t+1})\big) \\ &\leq \frac{\eta}{2} \| \nabla_t \|_\star^2 + \frac{\eta}{2} \| x_t - W_{t+1} \|^2 + \frac{1}{\eta}\big(D_F(x^\star, x_t) - D_F(x^\star, W_{t+1}) - D_F(x_t, , W_{t+1})\big) \\ &\leq \frac{\eta}{2} \| \nabla_t \|_\star^2 + \frac{1}{\eta}\big(D_F(x^\star, x_t) - D_F(x^\star, W_{t+1})\big) \end{aligned}$$

Note that the algorithm 1-③, we obtain

$$\begin{aligned} Reg_T &= \sum_t LHS_t \\ &\leq \frac{\eta}{2}\sum_{t=1}^{T} \| \nabla_t \|_\star^2 + \frac{1}{\eta}\sum_{t=1}^{T}\big(D_F(x^\star, x_t) - D_F(x^\star, W_{t+1})\big) \\ &\leq \frac{\eta}{2}\sum_{t=1}^{T} \| \nabla_t \|_\star^2 + \frac{1}{\eta}\sum_{t=1}^{T}\big(D_F(x^\star, x_t) - D_F(x^\star, x_{t+1})\big) \\ &\leq \frac{\eta}{2}T\cdot B^2 + \frac{1}{\eta}\big(D_F(x^\star, x_1) - D_F(x^\star, x_T)\big) \\ &\leq \frac{\eta}{2}T\cdot B^2 + \frac{1}{\eta}\big(F(x^\star) - F(x_1)\big) \\ &\leq \frac{\eta}{2}T\cdot B^2 + \frac{1}{\eta}\sup_{x\in D} F(x) \end{aligned}$$

The last 2nd line of the above proof is because of lemma 2-③. If we choose $\eta = \sqrt{\frac{2\sup_{x\in D} F(x)}{T\cdot B^2}}$, we then prove the theorem.

□

If we use neg-entropy function here we will obtain the same regret bound ($O(T\ln n)$) in projected GD as the MW algorithm does.

1.3 The Equivalence of FTRL and OMD

In this subsection we first introduce the FTRL algorithm which develops regularization algorithms for attaining low regret. [2] In line with the Follow the Leader algorithm, we give the name Follow the Regularized Leader to the following family of algorithms 2.

Algorithm 2: Follow the Regularized Leader (FTRL)

Given $\eta > 0$, previous loss function $f_t(x)$ and Mirror map F;
for $t = 1, ..., T$ **do**
  $x_{t+1} \leftarrow \arg\min_x \left(\eta \sum_{s=1}^{t} f_s(x) + F(x)\right)$;

Now we will show the equivalence of FTRL and OMD covered in class.

Proof of Equivalence of FTRL and OMD. Another interpretation of Online Mirror Descent is as following. Note that $\nabla F(W_{t+1}) = \nabla F(x_t) - \eta \nabla_t$, we obtain

$$\begin{aligned}
x_{t+1} &= \arg\min_{y \in D} D_F(y, W_{t+1}) \\
&= \arg\min_{y \in D} \Big(\big(F(y) - F(W_{t+1})\big) - \big\langle \nabla F(W_{t+1}), y - W_{t+1} \big\rangle \Big) \\
&= \arg\min_{y \in D} \Big(F(y) - \big\langle \nabla F(W_{t+1}), y - W_{t+1} \big\rangle \Big) \\
&= \arg\min_{y \in D} \Big(+ \eta \langle \nabla_t, y \rangle + F(y) - \big\langle \nabla F(x_t), y - x_t \big\rangle - F(x_t) \Big) \\
&= \arg\min_{y \in D} \Big(+ \eta \langle \nabla_t, y \rangle + D_F(y, x_t) \Big)
\end{aligned}$$

□

Here we can view F as a regularizer which means we do not want to go too far.

2 Multi-armed Bandit Problem

Reinforcement learning policies face the exploration versus exploitation dilemma, i.e. the search for a balance between exploring the environment to find profitable actions while taking the empirically best action as often as possible. A popular measure of a policy's success in addressing this dilemma is the regret, that is the loss due to the fact that the globally optimal policy is not followed all the times. One of the simplest examples of the exploration/exploitation dilemma is the multi-armed bandit problem.[1]

2.1 Introduction

In its most basic formulation, a K -armed bandit problem is defined by random variables $X_{i,n}$ for $1 \leq i \leq K$ and $n \geq 1$, where each i is the index of a gambling machine (i.e., the arm of a bandit). Successive plays of machine i yield rewards $X_{i,1}, X_{i,2}, ...$ which are independent and identically distributed according to an unknown law with unknown expectation μ_i. Independence also holds for rewards across machines; i.e., $X_{i,s}$ and $X_{j,t}$ are independent (and usually not identically distributed) for each $1 \leq i < j \leq K$ and each $s, t \geq 1$.

A policy, or allocation strategy, A is an algorithm that chooses the next machine to play based on the sequence of past plays and obtained rewards. Let $T_i(n)$ be the number of times machine i

has been played by A during the first n plays. Then the regret of A after T plays is defined by

$$Reg_T = \mu^\star \cdot T - \sum_{j=1}^{K} \mu_j \cdot \mathbb{E}[T_j(T)],$$

where $\mu^\star = \max_i \mathbb{E}[x_i] = \max_i \mu_i$.

2.2 Upper Confidence Bound (UCB) for N-armed Bandit Problem

Before discussing the Upper Confidence Bound (UCB) of multi-armed bandit problem, we first introduce the Chernoff bound, a not tight but useful bound.

Theorem 5 (Chrnoff bound) *For $X_1, ...X_n \in [0,1]$, let $X = \sum_i X_i$, $\mu = \sum_i \mathbb{E}[X_i]$. There exists*

$$\begin{cases} Pr[X \leq (1-\delta)\mu] \leq e^{-\delta^2\mu/2} \\ Pr[X \geq (1+\delta)\mu] \leq e^{-\delta^2\mu/3} \end{cases}$$

When μ is large and δ is small, the probability is small.

Algorithm 3: UCB

```
Initially paly each machine once. for t = 1, 2, ..., T do
    n_j ← times machine j is played before t;
    play machine j which maxmizes UCB: X̄_j + sqrt(2 ln t / n_j);
```

Next time we will show the regret bound of UCB is

$$Reg_T \leq O\left(\sum_{i:\mu_i<\mu^\star} \frac{\ln T}{\Delta_i}\right) + O\left(\sum_{j=1}^{K} \Delta_j\right),$$

where Δ_i is the gap of machine i, $\Delta_i = \mu^\star - \mu_i$. The last term is a constant w.r.t T.

Here is something useful about UCB in next class: Let $C_{t.s} = \sqrt{2 \ln t/w}$, then we have the fact that

$$\begin{cases} Pr[\overline{X_s^\star} \leq \mu^\star - C_{t,s}] \leq t^{-4} \\ Pr[X_{i,s} \geq \mu_i + C_{t,s}] \leq t^{-4} \end{cases}$$

References

[1] Auer P, Cesa-Bianchi N, Fischer P. Finite-time Analysis of the Multiarmed Bandit Problem[J]. Machine Learning, 2002, 47(2-3):235-256.

[2] `www-stat.wharton.upenn.edu/~rakhlin/courses/stat991/papers/lecture_notes.pdf`