Online Optimization, Learning, and Games (O2LG)

Lesson 7: Online Convex Optimization - Part I

Vinh Thanh Ho*, Panayotis Mertikopoulos

*Faculté des Sciences et Techniques
Université de Limoges
vinh-thanh.ho@unilim.fr

ACSYON
Master of Applied Mathematics

Table of Contents

Online Convex Optimization

Sequence of events: Online convex optimization (OCO)

Require: convex action set $\mathcal{X}$; convex loss functions $\ell_n\colon \mathcal{X} \to \mathbb{R}$, $n = 1, 2, \ldots$

repeat At each epoch $n = 1, 2, \ldots$ **do**

Choose action $x_n \in \mathcal{X}$. ▷ action selection

Encounter loss function $\ell_n\colon \mathcal{X} \to \mathbb{R}$. ▷ nature plays

Incur cost $c_n = \ell_n(x_n)$. ▷ reward phase

Observe entire loss function ℓ_n or gradient $g_n = \nabla \ell_n(x_n)$ or cost $c_n = \ell_n(x_n)$ ▷ feedback phase

until end

Feature: discrete time, single player, continuous actions, exogenous losses.

- Feedback: **depends** (function-based, gradient-based, loss-based, ...)

Feedback

Types of feedback

From best to worst (more to less info):

- Full information: observe entire loss function $\ell_n \colon \mathcal{X} \to \mathbb{R}$ # deterministic function feedback
- First-order info, exact: observe (sub)gradient $g_n \in \partial \ell_n(x_n)$ # deterministic vector feedback
- First-order info, inexact: observe noisy estimate of g_n # stochastic vector feedback
- 0th-order info (bandit): observe only incurred cost $c_n = \ell_n(x_n)$ # deterministic scalar feedback

Feedback

The oracle model

A stochastic first-order oracle (SFO) for $g_n \in \partial \ell_n(x_n)$ is a random vector of the form

$$\hat{v}_n = g_n + U_n + b_n, \tag{SFO}$$

where U_n is zero-mean and $b_n = \mathbb{E}[\hat{v}_n \mid \mathcal{F}_n] - g_n$ is the bias of $\hat{v}_n$.

Table of Contents

Regret

Performance measured by the agent's regret (loss formulation):

$$\mathrm{Reg}(T) = \max_{p\in\mathcal{X}} \sum_{n=1}^{T} [\ell_n(x_n) - \ell_n(p)] = \sum_{n=1}^{T} \ell_n(x_n) - \min_{p\in\mathcal{X}} \sum_{n=1}^{T} \ell_n(p).$$

- No regret: $\mathrm{Reg}(T) = o(T)$.
- Adversarial framework: minimize regret against any given sequence $\{\ell_n\}_n$.
- Expected regret: $\mathbb{E}[\mathrm{Reg}(T)] = \mathbb{E}\left[\max_{p\in\mathcal{X}} \sum_{n=1}^{T} [\ell_n(x_n) - \ell_n(p)]\right].$

  Pseudo-regret: $\overline{\mathrm{Reg}}(T) = \max_{p\in\mathcal{X}} \mathbb{E}\left[\sum_{n=1}^{T} [\ell_n(x_n) - \ell_n(p)]\right].$
- $\overline{\mathrm{Reg}}(T) \le \mathbb{E}[\mathrm{Reg}(T)]$.

Table of Contents

Be The Leader (BTL)

- Suppose that the entire loss function ℓ_n is observed before playing x_n.
- Then the agent can try to **Be The Leader (BTL)** by the following scheme:

$$x_n \in \arg\min_{x \in \mathcal{X}} \sum_{k=1}^{n} \ell_k(x). \tag{BTL}$$

Regret of BTL

☞ Under (BTL), the learner incurs $\operatorname{Reg}(T) \leq 0$. # Why?

...unrealistic

Follow The Leader (FTL)

- Suppose that the entire loss function ℓ_n is observed after playing x_n.
- Then the agent can try to **Follow The Leader (FTL)**

$$x_{n+1} \in \arg\min_{x \in \mathcal{X}} \sum_{k=1}^{n} \ell_k(x). \tag{FTL}$$

Does (FTL) lead to no regret?

Template bound for FTL

FTL regret bound

For all $p \in \mathcal{X}$, the regret of (FTL) can be bounded as

$$\mathrm{Reg}_p(T) = \sum_{n=1}^{T} [\ell_n(x_n) - \ell_n(p)] \leq \sum_{n=1}^{T} [\ell_n(x_n) - \ell_n(x_{n+1})]. \tag{1}$$

Proof.

Prove FTL regret bound (1). □

FTL against quadratic losses

Test (FTL) in an **Online Quadratic Optimization** (**OQO**) problem:

$$\ell_n(x) = \frac{1}{2}\|x - p_n\|^2 \quad \text{for some sequence of center points } p_n,\ n = 1, 2, \ldots \tag{OQO}$$

Regret of FTL in quadratic problems

☞ Assume: (FTL) is run against (OQO) with $\sup_n \|p_n\| \leq R$.

✓ Then:

$$\mathrm{Reg}(T) \leq 4R^2(1 + \log T). \tag{2}$$

Proof.

Prove the regret bound (2). Note that $\sum_{n=1}^{T} \frac{1}{n} \leq 1 + \int_1^T \frac{1}{t} dt.$ □

FTL against linear losses

Test (FTL) in an **Online Linear Optimization** (**OLO**) problem:

$$\ell_n(x) = \langle w_n, x \rangle \quad \text{for some sequence of loss vectors } w_n,\ n = 1, 2, \ldots \tag{OLO}$$

Chasing the leader

- Assume that $\mathcal{X} = [-1, 1]$. (FTL) is run against (OLO) with $w_1 = -\frac{1}{2}$ and $w_n = (-1)^n$ for $n \geq 2$.
- What is the incurred regret?

Follow The Regularized Leader (FTRL)

Add a fictitious "day zero loss" $\Longrightarrow$ Follow The Regularized Leader (FTRL)

$$x_{n+1} = \arg\min_{x \in \mathcal{X}} \left\{ \sum_{k=1}^{n} \ell_k(x) + \underbrace{\lambda h(x)}_{\text{“}\ell_0(x)\text{”}} \right\} \tag{FTRL}$$

where

- regularization function $h : \mathcal{X} \to \mathbb{R}$ is strongly convex. # $h - (K_h/2)\|\cdot\|^2$ convex for some $K_h > 0$
- regularization weight $\lambda > 0$ can be tuned.

Main idea: Use regularization $\Longrightarrow$ Achieve stability $\Longrightarrow$ Get less regret.

Shalev-shwartz et al. 2006, Shalev-Shwartz 2012

Example 1: Euclidean regularization

- Setup: linear losses $\ell_n(x) = \langle w_n, x \rangle$.
- Regularizer:

$$h(x) = \tfrac{1}{2}\|x\|^2.$$

- Algorithm:

$$x_{n+1} = x_n - \frac{1}{\lambda} w_n. \tag{3}$$

Prove the update step (3) of Algorithm (FTRL).

- Euclidean regularization + linear losses ($w_n = \nabla \ell_n(x_n)$) $\Longrightarrow$ *gradient descent:*

$$x_{n+1} = x_n - \underbrace{\eta}_{1/\lambda} \nabla \ell_n(x_n). \tag{GD}$$

Example 2: Entropic regularization

- Setup: $\mathcal{X} = \Delta(\mathcal{A})$, linear payoffs $u_n(x) = \langle v_n, x \rangle$.
- Regularizer:

$$h(x) = \sum_{a \in \mathcal{A}} x_a \log x_a.$$

- Algorithm:

$$x_{a,n+1} = \frac{\exp(-\sum_{k=1}^{n} v_{a,k}/\lambda)}{\sum_{a' \in \mathcal{A}} \exp(-\sum_{k=1}^{n} v_{a',k}/\lambda)}. \tag{4}$$

Prove the update step (4) of Algorithm (FTRL).

- Linear payoffs + Entropic regularization $\implies$ *exponential weights:* # Why?

$$y_{n+1} = y_n - \underbrace{\eta}_{1/\lambda} v_n \text{ and } x_{n+1} = \underbrace{\Lambda(y_{n+1})}_{\text{logit map}}. \tag{EW}$$

Template bound for FTRL

FTRL regret bound

For all $p \in \mathcal{X}$, the regret of (FTRL) can be bounded as

$$\mathrm{Reg}_p(T) \leq \lambda[h(p) - h(x_1)] + \sum_{n=1}^{T}[\ell_n(x_n) - \ell_n(x_{n+1})]. \tag{5}$$

Proof.

Prove the regret bound (5). □

Variability bound for FTRL

Variability of FTRL

☞ Assume that h is K_h-strongly convex; each ℓ_n is G_n-Lipschitz continuous.

✓ Then:

$$\ell_n(x_n) - \ell_n(x_{n+1}) \leq \frac{G_n^2}{\lambda K_h}. \tag{6}$$

Task 1

Prove the bound (6).

Regret of FTRL

Theorem 1 (Shalev-shwartz et al. 2006, Shalev-Shwartz 2012)

☞ *Assume: h is K_h-strongly convex; each ℓ_n is G-Lipschitz continuous.*

✓ *Then, for all $p \in \mathcal{X}$,* (FTRL) *enjoys the regret bound*

$$\mathrm{Reg}_p(T) \le \lambda[h(p) - \min h] + \frac{G^2}{\lambda K_h} T. \tag{7}$$

Task 2

Prove the regret bound (7).

Regret of FTRL

Corollary 1

With assumptions in Theorem 1,

$$R_h = \max h - \min h \quad \text{and} \quad \lambda = G\sqrt{T/(2K_h R_h)},$$

(FTRL) enjoys the regret bound

$$\mathrm{Reg}(T) \leq G\sqrt{(2R_h/K_h)\,T} = \mathcal{O}(\sqrt{T}).$$

Remarks:

- The bound is tight in T.
- Requires **full information** and tuning in terms of T.

Summary

This lesson

- Online Convex Optimization
- Learning with full information

Next lesson

- Learning with gradient feedback

References

[1] Shai Shalev-Shwartz. Online Learning and Online Convex Optimization. In: *Foundations and Trends® in Machine Learning* 4.2 (2012), pp. 107–194 (cited at slides -7, -2).

[2] Shai Shalev-shwartz and Yoram Singer. Convex Repeated Games and Fenchel Duality. In: *Advances in Neural Information Processing Systems*. Ed. by B. Schölkopf, J. Platt, and T. Hoffman. Vol. 19. MIT Press, 2006 (cited at slides -7, -2).