Online Optimization, Learning, and Games (O2LG)
Lesson 8: Online Convex Optimization - Part II

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

1. Learning with gradient feedback
2. Dual Averaging
3. Regret under Dual Averaging

Recall: 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 $\ell_n : \mathcal{X} \to \mathbb{R}$ # 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

Recall: Follow The Regularized Leader (FTRL)

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

Follow The Linearized Leader (FTLL)

Can we relax the full information requirement of (FTRL)?

- Replace ℓ_n with first-order surrogate $\hat{\ell}_n$ defined by:

$$\hat{\ell}_n(x) = \ell_n(x_n) + \langle g_n, x - x_n \rangle, \qquad g_n \in \partial \ell_n(x_n).$$

- Plugging into (FTRL) and setting $\lambda := 1/\eta$, yield **Follow The Linearized Leader (FTLL)**:

$$x_{n+1} = \arg\min_{x \in \mathcal{X}} \left\{ \eta \sum_{k=1}^{n} \langle g_k, x \rangle + h(x) \right\}. \tag{1}$$

How to obtain the update step (1) of (FTLL)?

Table of Contents

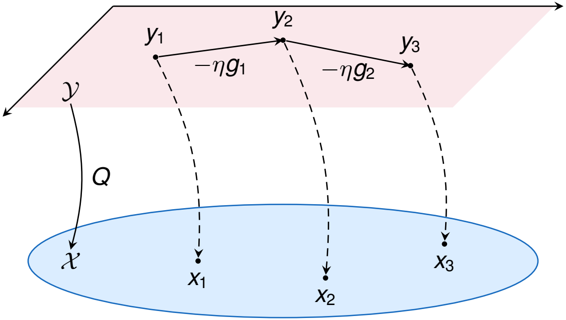
1. Learning with gradient feedback
2. Dual Averaging
3. Regret under Dual Averaging

Dual Averaging (DA)

Dual Averaging (DA) formulation of (FTLL): Nesterov 2009; Xiao 2010

$$\begin{aligned} y_{n+1} &= y_n - \eta g_n, \\ x_{n+1} &= Q(y_{n+1}), \end{aligned} \tag{DA}$$

where $Q(y) = \arg\max_{x \in \mathcal{X}} \{\langle y, x \rangle - h(x)\}$ is the mirror map associated to h.



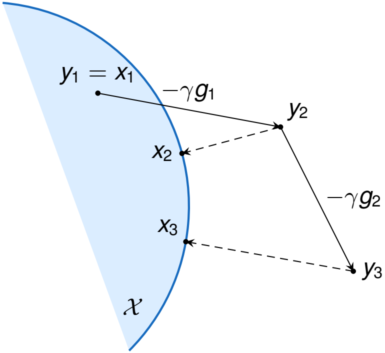
Example: Online Gradient Descent (OGD)

Special case when the regularizer $h(x) = \frac{1}{2}\|x\|_2^2 \rightsquigarrow$ **Online Gradient Descent (OGD)**

$$y_{n+1} = y_n - \eta g_n, \qquad x_{n+1} = \Pi_{\mathcal{X}}(y_{n+1}), \tag{OGD}$$

where $\Pi_{\mathcal{X}}$ is the projection of a point onto the set $\mathcal{X}$.

How to obtain the update step of (OGD)?



Online Mirror Descent (deep dive)

- Gradient signals $\{g_n\}_n$ enter (DA) unweighted / unadjusted.
- Variable weights $\rightsquigarrow$ "lazy", primal-dual variant of Online Mirror Descent (OMD):

$$\begin{aligned} y_{n+1} &= y_n - \eta_n g_n, \\ x_{n+1} &= Q(y_{n+1}). \end{aligned} \tag{OMD_{lazy}}$$

- Primal-primal ("eager") variant of (OMD_{lazy}):

$$x_{n+1} = P_{x_n}(-\eta_n g_n). \tag{OMD}$$

The **Bregman proximal mapping** P is defined as

$$P_x(w) := \arg\min_{x' \in \mathcal{X}} \{\langle w, x - x' \rangle + D(x', x)\}$$

where $D(z, v) = h(z) - h(v) - \langle \nabla h(v), z - v \rangle$ is the Bregman divergence of h.

Table of Contents

1. Learning with gradient feedback
2. Dual Averaging
3. Regret under Dual Averaging

Regret under Dual Averaging

- **Gradient trick:** # linear model

$$\ell_n(x_n) - \ell_n(p) \leq \langle g_n, x_n - p \rangle \quad \text{for all } p \in \mathcal{X}.$$

- **Energy function:**

$$F_n = h(p) + h^*(y_n) - \langle y_n, p \rangle,$$

where $h^*(y) = \max_{x \in \mathcal{X}} \{\langle y, x \rangle - h(x)\}$ is the potential of $Q \rightsquigarrow \nabla h^* = Q$.

The function h^* enjoys the following inequality: # Pages 227-228, Nesterov 2009

$$h^*(s + \delta) \leq h^*(s) + \langle \delta, \nabla h^*(s) \rangle + \frac{1}{2K_h} \|\delta\|^2, \ \forall s, \delta \in \mathcal{Y}. \tag{2}$$

Regret under Dual Averaging

- Template inequality: # prove the inequality

$$F_{n+1} \leq F_n - \eta\langle g_n, x_n - p\rangle + \frac{\eta^2}{2K_h}\|g_n\|^2.$$

- Rearrange & telescope: # how to build the regret

$$\mathrm{Reg}(T) \leq \frac{R_h}{\eta} + \frac{\eta}{2K_h}\sum_{n=1}^{T} G_n^2,$$

where $R_h = \max h - \min h$; ℓ_n is G_n-Lipschitz; $\|g_n\| \leq G_n$ for all n. # Shalev-Shwartz 2012

Regret under Dual Averaging

Theorem 1 (Shalev-Shwartz 2012)

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

$$R_h = \max h - \min h; \quad \eta = G^{-1}\sqrt{2K_hR_h/T}.$$

✓ *Then:* (DA) *enjoys the regret bound*

$$\mathrm{Reg}(T) \leq G\sqrt{\frac{2R_hT}{K_h}}.$$

Task

Prove this theorem.

Summary

This lesson

- Learning with gradient feedback
- Dual Averaging
- Regret under Dual Averaging

Next lesson

- Learning with stochastic gradients
- ...

References

[1] Yurii Nesterov. Primal-dual subgradient methods for convex problems. In: *Mathematical Programming* 120.1 (June 2009), pp. 221–259 (cited at slides -7, -3).

[2] 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 -2, -1).

[3] Lin Xiao. Dual Averaging Methods for Regularized Stochastic Learning and Online Optimization. In: *Journal of Machine Learning Research* 11.88 (2010), pp. 2543–2596 (cited at slide -7).