Lecture 01: Introduction to Online Learning

Lecturer: John Lazarsfeld

September 16, 2025

Abstract

Fundamentals of online learning: prediction with expert advice, online convex optimization, external regret, and motivating examples. Introduction to Online Gradient Descent (OGD), and a proof of its regret guarantee.

1 Online Learning Problem Setting

1.1 Introduction

The *online learning* framework captures sequential decision making problems in adaptive, non-stationary environments. In its simplest form, online learning consists of a *learner* tasked with choosing *actions* over a sequence of rounds. In each round, after choosing an action, the learner incurs a loss and observes feedback from the enivornment. The learner's goal is to minimize its total incurred loss over time, using the prior feedback to inform its future actions.

Overview of lecture. The goal of this lecture is to formally introduce this problem setting. We do this in two parts: first, by describing the special *prediction with expert advice* setting and the notion of *external regret*, and then by introducing the more general *online convex optimization* setting (of which the former is a special case). Special attention is given to motivating and distinguishing between the distinct components of the online learning setting: the structure of the loss functions, the feedback available to the learner, and the choice of regret benchmark. We will then introduce the fundamental Online Gradient Descent (OGD) algorithm and prove its regret guarantee.

1.2 Prediction with Expert Advice

Overview of setting. The *Prediction with Expert Advice* setting (henceforth, the *experts* setting) was introduced by Littlestone and Warmuth (1994) and is an instance of the sequential decision-making scenario described above with a *finite action set* and *linear loss functions*. In this setting, we assume the learner chooses a *distribution* x_t over the actions at each round t. The goal of the learner is to then use its past observation to minimize its total *expected loss* over time.

Some notation. To make this setting precise, we briefly recall some notations that will be used throughout. Let $[n] = \{1, 2, \dots, n\}$. Let $\Delta_n = \{x \in \mathbb{R}^n : x_i \geq 0 \text{ for all } i, \text{ and } \sum_{i=1}^n x_i = 1\}$ denote the probability simplex over $[n]$. For $u, v \in \mathbb{R}^n$, we denote the ℓ_2 inner product $\langle u, v \rangle = \sum_{i=1}^n u_i v_i$, and the ℓ_2-norm by $\|u\| = \sqrt{\langle x, x \rangle}$.

The experts setting is then formally defined as follows:

Setting 1.1 (Experts setting). *At each round $t \in [T]$:*

1. *The learner chooses a distribution $x_t \in \Delta_n$.*
2. *An adversary/nature chooses a loss vector $\ell_t \in \mathbb{R}^n$.*
3. *The learner observes ℓ_t and incurs cost $\langle \ell_t, x_t \rangle$.*

Remark 1. We make several important remarks on this setting:

1. **Adversarially-chosen losses**: In step (2), we make *no assumptions* on how the loss vector ℓ_t is *generated*. In particular, each ℓ_t could be adversarially chosen and depend on x_t (and

importantly, the learner must choose x_t without knowledge of ℓ_t). On the other hand, the sequence of loss vectors may also have some nice underlying (non-adversarial) structure. The model is robust to these different scenarios.

2. **Full feedback model**: In step (3), the learner observes the loss of all n actions via the vector ℓ_t. We refer to this as the *full feedback* model.

Measuring performance via *regret*. The goal of the learner is to minimize its total incurred cost over time. This quantity is given by the sum $\sum_{t=1}^{T}\langle \ell_t, x_t\rangle$. However, in online learning settings, we will measure the performance of a learning algorithm via *regret*. Roughly speaking, the regret of an algorithm is the difference between its total incurred loss over time, and the cumulative loss of some fixed benchmark policy on the same sequence of losses. The most fundamental benchmark is the policy that plays a fixed action at every round, and this leads to the definition of *external regret*:

Definition 2 (External regret in experts setting). *Let $\mathcal{A}$ be an online learning algorithm. Consider an instance of Setting 1.1, where $\mathcal{A}$ outputs distributions $\{x_t\}$ and the adversary/nature outputs loss vectors $\{\ell_t\}$. Then the external regret of $\mathcal{A}$ over T rounds is*

$$\mathrm{Reg}_{\mathcal{A}}(T) := \sum_{t=1}^{T}\langle \ell_t, x_t\rangle \;-\; \min_{x\in\Delta_n}\sum_{t=1}^{T}\langle \ell_t, x\rangle \;. \tag{1}$$

In online learning settings, our goal is to design algorithms that guarantee $\mathrm{Reg}_{\mathcal{A}}(T)$ grows *sublinearly in T* against every sequence of losses. More formally:

Definition 3 (No Regret)**.** *Consider an online learning algorithm $\mathcal{A}$ for the experts setting. We say that $\mathcal{A}$ is a **no-(external)-regret** algorithm if $\mathrm{Reg}_{\mathcal{A}}(T) = o(T)$ is sublinear. Equivalently, $\frac{\mathrm{Reg}_{\mathcal{A}}(T)}{T} = o(1)$.*

The no-(external)-regret definition implies that on average (over all rounds) the difference between the incurred cost of the learner and the incurred cost of the *best fixed action in hindsight* is zero. In general, algorithms obtaining $O(\sqrt{T})$ regret in this setting are optimal (we will see lower bounds of this order in future lectures).

Remark 4 (On (external) regret). We make several additional assumptions on regret:

1. **Computability of the comparator**:
   First, we will usually refer to a fixed action $x^\star \in \mathrm{argmin}_{x\in\Delta_n}\sum_{t=1}\langle \ell_t, x\rangle$ as a *comparator*. We call $\min_{x\in\Delta_n}\sum_{t=1}^{T}\langle \ell_t, x\rangle = \sum_{t=1}^{T}\langle \ell_t, x^\star\rangle$ the *loss of the comparator*.

   Observe this comparator can only be computed *in hindsight*. In other words, the fixed distribution $x \in \Delta_n$ that minimizes $\sum_{t=1}^{T}\langle \ell_t, x\rangle$ can only be computed after all rounds $t = 1, \ldots, T$ have passed. In the online learning setting, the future loss vectors are unknown to the learner. Thus, the learner cannot simply solve the *offline problem* $\min_{x\in\Delta_n}\sum_{t=1}^{T}\langle \ell_t, x\rangle$. However, ensuring $\mathrm{Reg}_{\mathcal{A}}(T) = o(T)$ means that the cost of the learner's action choices "converge" (on average) to the cost of the best fixed distribution in hindsight.

2. **Standard assumption of bounded losses**:
   In the experts setting with linear loss vectors, observe that the best comparator $x^\star \in \mathrm{argmin}_{x\in\Delta_n}\sum_{t=1}\langle \ell_t, x\rangle$ is a *point-mass distribution* (e.g., a vertex of the simplex Δ_n and standard basis vector $e_i \in \mathbb{R}^n$).

   While the setup of Step (2) in Setting 1.1 placed no assumptions on the loss vector $\ell_t \in \mathbb{R}^n$, in order to obtain meaningful (sublinear) regret bounds, we must place some boundedness conditions on ℓ_t. In particular, we will usually assume for all $t \in [T]$ that $\ell_t \in [-c, c]^n$ for some constant $c > 0$. This assumption ensures that the incurred cost per-round is also uniformly bounded, and allows for a more fair comparison of an algorithm's performance

to the best comparator.

Example 5 (Portfolio selection). For clarity, we give the following example of the experts setting and external regret definition. Consider a *portfolio selection* task, where in each timestep, a learner must allocate a firm's funds over three stocks (e.g., Google, Microsoft, Apple) by way of choosing an allocation distribution $x_t = (x_{t,1}, x_{t,2}, x_{t,3}) \in \Delta_3$ over the three stocks. After each day, the learner observe a loss vector $\ell_t = (\ell_{t,1}, \ell_{t,2}, \ell_{t,3})$ specifying each stock's change in share price (with negative loss values indicating a rise in share price).

Consider the learner chooses the following sequence of distibutions observes the following loss vectors:

- **Day 1**: $x_1 = (1/3, 1/3, 1/3)$ and $\ell_1 = (-9, -6, -1)$; Thus $\langle \ell_1, x_1 \rangle = -6$.
- **Day 2**: $x_2 = (2/3, 1/6, 1/6)$ and $\ell_2 = (-3, -9, -3)$; Thus $\langle \ell_2, x_2 \rangle = -4$.
- **Day 3**: $x_3 = (1/6, 2/3, 1/6)$ and $\ell_3 = (-12, -3, -3)$; Thus $\langle \ell_3, x_3 \rangle = -4.5$.

We compute the cost of the best comparator $\min_{x \in \Delta_3} \langle x, \sum_{t=1}^3 \ell_t \rangle = \min\{-24, -18, -7\} = -24$, which is minimized by the first stock Google. Meanwhile, the total cost incurred by the learner's allocation choices is $\sum_{t=1}^3 \langle x_t, \ell_t \rangle = -14.5$. Thus the regret of the learner is $-14.5 - (-24) = 9.5$.

1.3 Online (Convex) Optimization

The *experts* setting consists of linear loss functions (loss vectors) over the n-dimensional simplex. We now introduce the more general setting of *Online Convex Optimization (OCO)*, which dates back to Gordon (1999) and Zinkevich (2003). Here, the losses can now be arbitrary convex functions, and the action space is a general convex subset of $\mathbb{R}^n$. For this, we first recall some important definitions from convex analysis:

Refresher on convexity. We recall several key notions from convex analysis. For a more thorough treatment, see Hazan et al. (2016, Section 2.1) and Orabona (2019, Section 2.1.1).

(a) **Convex sets and functions**:

Definition 6. *A set $\mathcal{X} \subset \mathbb{R}^n$ is convex if for all $x, x' \in \mathcal{X}$ and $\lambda \in [0,1]$: $\lambda x + (1-\lambda)x' \in \mathcal{X}$.*

Definition 7. *A function $f : \mathcal{X} \to \mathbb{R}$ is convex if for all $x, x' \in \mathcal{X}$ and $\lambda \in [0,1]$:*

$$f((1-\lambda)x + \lambda x') \leq (1-\lambda)f(x) + \lambda f(x') \,.$$

Lemma 8. *Suppose $f : \mathcal{X} \to \mathbb{R}$ is a differentiable function. Then f is convex if and only if for all $x, x' \in \mathcal{X}$:*

$$f(x') - f(x) \geq \langle \nabla f(x), x' - x \rangle \,.$$

We now define the OCO setting formally:

Setting 1.2 (Online Convex Optimization). *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be a convex set. At each round $t \in [T]$:*

1. *The learner chooses $x_t \in \mathcal{X}$.*
2. *An adversary/nature chooses a convex loss function $f_t : \mathcal{X} \to \mathbb{R}$.*
3. *The learner observes f_t and incurs cost $f_t(x_t)$.*

In the OCO setting, the definition of *external regret* is the natural generalization of Definition 2 for the experts setting:

Definition 9 (External regret in OCO setting). *Let $\mathcal{A}$ be an online learning algorithm. Consider an instance of Setting 1.1, where $\mathcal{A}$ outputs actions $\{x_t\}$ and the adversary/nature outputs loss functions*

$\{f_t\}$. *Then the external regret of* $\mathcal{A}$ *over* T *rounds is*

$$\mathrm{Reg}_{\mathcal{A}}(T) := \sum_{t=1}^{T} f_t(x_t) \; - \; \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x) \; . \tag{2}$$

Remark 10. We make several remarks on the setting and external regret definition:

1. **On the decision set $\mathcal{X}$:**
   When $\mathcal{X} \subset \mathbb{R}^n$, we will usually assume that the decision set $\mathcal{X}$ is convex and compact (e.g., closed and bounded). For example in the experts setting, $\mathcal{X} := \Delta_n$ satisfies these assumptions.

2. **Full feedback model**:
   In Step (3) of the setting, we assume the full feedback model where the learner observes the loss function f_t. Given that the learner knows their own action choise x_t, this feedback is equivalent to having access to the gradient $\nabla f_t(x_t) \in \mathbb{R}^n$.

3. **Connection between external regret and average-iterate convergence rates**:
   When the loss functions are static, meaning $f_t = f$ for all $t \in [T]$, then upper bounds on the external regret $\mathrm{Reg}_{\mathcal{A}}(T)$ correspond to *average-iterate* convergence rates for minimizing f. To see this, let $\widetilde{x}_t := \frac{1}{T}\sum_{t=1}^{T} x_t$ denote the average iterate output by an algorithm $\mathcal{A}$. As f is convex, by Jensen's inequality[1] we have $\frac{1}{T}\sum_{t=1}^{T} f(x_t) \geq f(\widetilde{x}_t)$. It follows that

$$\frac{1}{T}\mathrm{Reg}_{\mathcal{A}}(T) = \frac{1}{T}\sum_{t=1}^{T} f(x_t) - \min_{x \in \mathcal{X}} f(x) \geq f(\widetilde{x}_t) - \min_{x \in \mathcal{X}} f(x) \; .$$

4. **Linear losses are "worst-case" for OCO**:
   Every sequence of (adversarially-chosen) convex loss functions $\{f_t\}$ "reduces" to a sequence of *linear* loss function (loss vectors). To see this, let $x^\star \in \mathrm{argmin}_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x)$. Then by convexity of f (Lemma 8), we have for each $t \in [T]$ that

$$f_t(x_t) - f_t(x^\star) \leq \langle x_t - x^\star, \nabla f_t(x_t) \rangle = \langle x_t, \nabla f_t(x_t) \rangle - \langle x^\star, \nabla f_t(x_t) \rangle \; .$$

   Writing $\ell_t := \nabla f_t(x_t)$ for each t and recalling the definition of $x^\star$, observe then that

$$\mathrm{Reg}(T) \; = \; \sum_{t=1}^{T} f_t(x_t) - f_t(x^\star) \; \leq \; \sum_{t=1}^{T} \langle x_t, \ell_t \rangle - \langle x^\star, \ell_t \rangle \; \leq \; \sum_{t=1}^{T} \langle x_t, \ell_t \rangle - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} \langle x, \ell_t \rangle \; . \tag{3}$$

   Observe that this final term is essentially the definition of regret from the experts setting (Definition 2)[2]. Thus, if we can bound an algorithm's regret against any adversarially-chosen sequence of linear loss vectors, then equation (3) suggests this bound also applies for the more general case of convex loss functions. For this reason, we will often focus on the linear case when analyzing the regret of online learning algorithms. For more discussion, see (Orabona, 2019, Section 2.3).

1.4 Spectrum of Online Learning Settings

So far, we have introduce the standard OCO setting with (1) possibly adversarially-chosen loss functions, (2) full feedback (equivalently, gradient feedback), and (3) using external regret as a measure of performance (e.g., comparator class of fixed actions).

Throughout the course, we study variations of the setting along with each of these three components. To briefly mention a few of these variations:

[1] This is the multi-point extension of the first-order convexity condition in Lemma 8.

[2] The difference in (3) compared to Definition 2 is that the action set $\mathcal{X} \subseteq \mathbb{R}^n$ is not necessarily the simplex Δ_n.

- **Structure in loss functions**: the sequence of loss functions may be adaptative, but in a non-adversarial manner (e.g., when generated within a multi-agent games setting). When the loss function has less variability over time, algorithms may perform better than for worst case, adversarial sequences.
- **Feedback**: the feedback available to the learner may be limited or noisy. For example, the learner might only obtain *bandit* feedback of the form $f_t(x_t) \in \mathbb{R}$. Or, they might obtain a noisy or biased estimate of $\nabla f_t(x_t)$ (or $f_t(x_t)$). In these cases, we may expect algorithms to perform worse compared to the full-feedback setting.
- **Comparator class**: The set of comparators in the regret definiton may extend beyond fixed actions. Instead, we may consider richer sets of comparators (e.g., that are time-varying) that correpsond to refined notions of regret. Minimizing these other notions of regret have consequences for learning equilibria in multi-player games settings.

2 Online Gradient Descent

2.1 Overview

We now discuss a more general online learning algorithm for the OCO setting: *Online Gradient Descent*. Online Gradient Descent (OGD) is the natural online analog of standard Gradient Descent: at every round the learner iteratively updates its action choice in a greedy manner, moving (roughly speaking) in the direction that maximally reduces its most recent loss.

Under a standard stepsize parameter setting, and assuming a *boundedess* property on the feedback observed by the learner, we will state and develop the proof of a *sublinear* regret bound for (OGD).

Refresher on Lipschitzness and projections. We first recall a few additional concepts from convex analysis that are needed for analyzing OGD (see Hazan et al. (2016, Section 2.1)).

(a) **Projections onto convex sets**:

Definition 11. *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be compact and convex. For $x \in \mathcal{X}$, define the projection operator*

$$\Pi_{\mathcal{X}}(x) := \operatorname{argmin}_{x' \in \mathcal{X}} \|x - x'\| .$$

Lemma 12. *Let $\mathcal{X} \subseteq \mathbb{R}^n$ be compact and convex. For any $x \in \mathbb{R}^n$ and $x' \in \mathcal{X}$:*

$$\|\Pi_{\mathcal{X}}(x) - x'\| \leq \|x - x'\| .$$

(b) **Lipschitzness and boundedness**:

Definition 13. *A function $f : \mathcal{X} \to \mathbb{R}$ is L-Lipschitz (for a constant $L > 0$) with respect to $\|\cdot\|$ if, for all $x, x' \in \mathcal{X}$:*

$$|f(x) - f(x')| \leq L \cdot \|x - x'\| .$$

Lemma 14. *A function $f : \mathcal{X} \to \mathbb{R}$ is L-Lipschitz if and only if $\|\nabla f(x)\| \leq L$ for all $x \in \mathcal{X}$.*

Online Gradient Descent algorithm. We now state the OGD algorithm:

In step (1) of the algorithm, we assume the learner observes the gradient feedback vector $\nabla f_t(x_t) \in \mathbb{R}^n$ in each round. Recall that this is equivalent to the full-feedback assumption of Setting 1.2 (c.f., Point 2 in Remark 10).

Algorithm 1 Online Gradient Descent (OGD) for OCO Setting

Input: Compact and convex set $\mathcal{X} \subset \mathbb{R}^n$; initialization $x_0 \in \mathcal{X}$; stepsize parameter $\eta > 0$.
for $t = 1, \dots, T$ **do**:
1. Play action $x_t \in \mathcal{X}$, and incur cost $f_t(x_t)$. Observe feedback $\nabla f_t(x_t) \in \mathbb{R}^n$.
2. Perform the update
$$x_{t+1} := \Pi_{\mathcal{X}}\left(x_t - \eta \nabla f_t(x_t)\right) . \tag{4}$$

end for

Regret guarantee for OGD. Under a time-invariant setting of the stepsize parameter η, we have the following, general regret bound for OGD:

Theorem 15. *Consider Setting 1.2 with loss functions $\{f_t\}$ and convex and compact decision set $\mathcal{X} \subseteq \mathbb{R}^n$. Let $x^\star \in \operatorname{argmin}_{x \in \mathcal{X}} \sum_{t=1}^T f_t(x)$. Then running Online Gradient Descent (Algorithm 1) with stepsize $\eta > 0$ achieves*

$$\mathrm{Reg}_{OGD}(T) \;=\; \sum_{t=1}^T f_t(x_t) - f_t(x^\star) \;\le\; \frac{\eta}{2}\Big(\sum_{t=1}^T \|\nabla f_t(x_t)\|^2\Big) + \frac{\|x_1 - x^\star\|}{2\eta} . \tag{5}$$

As a corollary, if the learner has access to a uniform bound on the gradient norms (equivalently, if the functions are uniformly Lipschitz), as well as a bound on the diameter of $\mathcal{X}$, then under an appropriate setting of η, the following sublinear regret bound is achievable:

Corollary 16. *Assume the setting of Theorem 15. Suppose the learner knows (i) a constant $L > 0$ such that $\|\nabla f_t(x_t)\| \le L$ for all $t \in [T]$, and (ii) a constant $D > 0$ such that $\|x - x'\| \le D$ for all $x, x' \in \mathcal{X}$. Set $\eta = \frac{D}{L\sqrt{T}}$. Then:*

$$\mathrm{Reg}_{OGD}(T) \;\le\; DL\sqrt{T} .$$

Remark 17. We make several additional remarks on the algorithm and regret guaratnee:

1. **Extending the anlaysis to subgradients:**
If the loss functions $\{f_t\}$ are not all differentiable, the OGD algorithm extends naturally to the case of using *subgradients*, and the regret guarantee remains the same. See (Orabona, 2019, Section 2.2).

2. **Time-varying step sizes:**
In the algorithm, we consider the case of a fixed, time-invariant stepsize η. Moreover, in Corollary 16, we obtain the final sublinear regret bound with a setting of η depending on the time-horizon T.

   In general, we may also want to consider stepsizes that vary with time (and that do not have a fixed dependence on the time-horizon T). Time-varying stepsize schedules decaying like $\eta_t = O(1/\sqrt{t})$ lead to similar regret guarantees as in Theorem 15 and Corollary 16. See Hazan et al. (2016, Theorem 3.1) and Orabona (2019, Theorem 2.13).

2.2 Proof of Theorem 15 and Corollary 16

We develop the proof in several steps:

(i) **Upper bound on regret via convexity.**

Fix $x^\star \in \operatorname{argmin}_{x \in \mathcal{X}} \sum_{t=1}^T f_t(x)$. Using the definition of $\mathrm{Reg}_{OGD}(T)$ and the convexity of the loss functions $\{f_t\}$ (in particular, Lemma 8 applied at each time t), we can write

$$\mathrm{Reg}_{OGD}(T) \;=\; \sum_{t=1}^T f_t(x_t) - f_t(x^\star) \;\le\; \sum_{t=1}^T \langle \nabla f_t(x_t), x_t - x^\star \rangle . \tag{6}$$

(ii) **Control the "instantaneous regret" terms in (6).**

We want to derive an upper bound on each term $\langle \nabla f_t(x_t), x_t - x^\star \rangle$ from the sum in (6). For this, we use the OGD update rule of x^{t+1} from (4) to write:

$$\|x_{t+1} - x^\star\|^2 \;=\; \|\Pi_{\mathcal{X}}\big(x_t - \eta \nabla f_t(x_t)\big) - x^\star\|^2 \,. \tag{7}$$

By the non-expansivity of the projection operator (Lemma 12) and by expanding the square, we can further write

$$\begin{aligned} \|\Pi_{\mathcal{X}}\big(x_t - \eta \nabla f_t(x_t)\big) - x^\star\|^2 \;&\leq\; \|x_t - \eta \nabla f_t(x_t) - x^\star\|^2 &(8)\\ &=\; \|x_t - x^\star\|^2 - 2\eta \langle \nabla f_t(x_t), x_t - x^\star \rangle + \eta^2 \|\nabla f_t(x_t)\|^2 \,. &(9)\end{aligned}$$

Combining equations (7) and (9) and rearranging, we find

$$\langle \nabla f_t(x_t), x_t - x^\star \rangle \;\leq\; \frac{\eta}{2}\|\nabla f_t(x_t)\|^2 + \frac{\|x_t - x^\star\|^2 - \|x_{t+1} - x^\star\|^2}{2\eta} \tag{10}$$

(iii) **Sum the terms in (10) and simplify.**

Summing the second term of (10) over all $t \in [T]$ yields a telescoping series, and thus

$$\sum_{t=1}^{T} \frac{\|x_t - x^\star\|^2 - \|x_{t+1} - x^\star\|^2}{2\eta} \;=\; \frac{\|x_1 - x^\star\|^2 - \|x_{T+1} - x^\star\|^2}{2\eta} \;\leq\; \frac{\|x_1 - x^\star\|^2}{2\eta} \,. \tag{11}$$

Here, the inequality comes from the non-negativity of $\|x_{T+1} - x^\star\|^2$.

Now combining expressions (11), (10), and (6), we conclude that

$$\mathrm{Reg}_{OGD}(T) \;\leq\; \sum_{t=1}^{T} \langle \nabla f_t(x_t), x_t - x^\star \rangle \;\leq\; \frac{\eta}{2}\Big(\sum_{t=1}^{T} \|\nabla f_t(x_t)\|^2\Big) + \frac{\|x_1 - x^\star\|^2}{2\eta} \,. \tag{12}$$

This proves Theorem 15.

(iv) **Proof of Corollary 16: instantiate η.**

Under the assumption that $\|\nabla f_t(x_t)\| \leq L$ for all $t \in [T]$, and that $\|x - x'\| \leq D$ for all $x, x' \in \mathcal{X}$, expression (12) can be further bounded by

$$\mathrm{Reg}_{OGD}(T) \;\leq\; \frac{\eta L^2 T}{2} + \frac{D^2}{2\eta} \,. \tag{13}$$

To obtain the sharpest bound on (13), we set η such that the two terms balance, and this yields a setting of

$$\frac{\eta L^2 T}{2} = \frac{D^2}{2\eta} \;\Longrightarrow\; \eta^2 = \frac{D^2}{L^2 T} \;\Longrightarrow\; \eta = \frac{D}{L\sqrt{T}} \,. \tag{14}$$

Plugging this setting of η into (13), we simplify and conclude

$$\mathrm{Reg}_{OGD}(T) \;\leq\; \frac{DL\sqrt{T}}{2} + \frac{DL\sqrt{T}}{2} \;=\; DL\sqrt{T} \,. \qquad \square$$

References

Geoffrey J Gordon. Regret bounds for prediction problems. In *Proceedings of the twelfth annual conference on Computational learning theory*, pages 29–40, 1999.

Elad Hazan et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Nick Littlestone and Manfred K Warmuth. The weighted majority algorithm. *Information and computation*, 108(2):212–261, 1994.

Francesco Orabona. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213*, 2019.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.