Online Learning in Games

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 1: Introduction to online learning

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-735 (Spring 2024)

License Information for Online Learning in Games Slides

- This work is released under a Creative Commons License with the following terms:
- **Attribution**
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees must give the original authors credit.
- **Non-Commercial**
  - The licensor permits others to copy, distribute, display, and perform the work. In return, licensees may not use the work for commercial purposes – unless they get the licensor's permission.
- **Share Alike**
  - The licensor permits others to distribute derivative works only under a license identical to the one that governs the licensor's work.
- Full Text of the License

Logistics

Credits	4
Lectures	Thursday 9:15-12:00 (CM011)
Practical hours	Thursday 9:15-12:00 starting 11th of April (CM011)
Prerequisites	Previous coursework in calculus, linear algebra, and probability is required. Familiarity with optimization is useful.
Grading	Preparation & presentation of a lecture given in week 14, 3-7th of June (cf., course book). Participation is mandatory during this week – please make sure you are available!
Moodle	`https://go.epfl.ch/OLIG`.
Course book	`https://edu.epfl.ch/coursebook/en/online-learning-in-games-EE-735`
LIONS	Stratis Skoulakis, Kimon Antonakopoulos, Thomas Pethick

Acknowledgements

These slides would not have been possible without the help of

Kimon Antonakopoulos, Thomas Pethick and Stratis Skoulakis

Outline of this lecture

- Offline minimization recap
- Online optimization
  - What is the setting?
  - How do we measure performance?
- Important special cases
  - The expert problem
  - Online path selection
  - Spam filtering
  - Portfolio management I
- Algorithms
  - The Hedge algorithm
  - Online gradient descent
  - Follow the regularized leader
- Lower bounds
- Online to offline
  - Online to batch conversion
  - Solving zero-sum games

(Offline) Convex optimization

Convex optimization

Given a convex and differentiable function $f : \mathcal{X} \mapsto \mathbb{R}$, we are interested in the following optimization problem

$$\min_{x \in \mathcal{X}} f(x),$$

where f is proper, closed, and twice-continuous differentiable without loss of generality.

Iterative methods (re-described in our convention)

For each round $t = 1, \ldots, T$, given the fixed (*offline*) optimization objective

- An *algorithm* selects an $x^t \in \mathcal{X}$.
- The *algorithm* **receives feedback**:
  - $f(x^t)$ Zero-order access;
  - $\nabla f(x^t)$ First-order access;
  - $\nabla^2 f(x^t)$ Second-order access;
- The algorithm gets evaluated on how small $f(x^T)$ is.

Examples:

- Gradient descent, i.e., $x^{t+1} = x^t - \gamma \nabla f(x^t)$, is a first-order method (γ is the step-size).
- Newton's method, i.e., $x^{t+1} = x^t - \nabla^2 f(x^t)^{-1} \nabla f(x^t)$, is a second-order method.

Online convex optimization (OCO)

- Proposed by Zinkevic et al. [27], OCO studies the twist when the **objective function** f changes over time.
  - Applications: (offline) convex optimization, online decision making, machine learning...

Online convex optimization (Zinkevic et al. [27])

At each round $t = 1, \dots, T$, where T is the time horizon,

- A *learner* selects an $x^t \in \mathcal{X}$.
- An *adversary* selects a function $f_t \in \mathcal{F} : \mathcal{X} \mapsto \mathbb{R}$.
- The *learner* **suffers** cost $f_t(x^t)$ and **receives feedback** $\nabla_t := \nabla f_t(x^t)$.

Remarks:

- The *learner* should select $x^t \in \mathcal{X}$ **solely based on** $\nabla_1, \dots, \nabla_{t-1}$ to minimize its overall cost:

$$\text{Learner's cost} \quad := \sum_{t=1}^{T} f_t(x^t).$$

- The *adversary* should not be all powerful, and hence, is restricted to a class of functions $\mathcal{F}$.
- Since the adversary selects $f_t(\cdot)$ *after* the learner' s selection $x^t \in \mathcal{X}$, competing the *best time-changing sequence*, $\sum_{t=1}^{T} \left(f_t(x^t) - \min_{x \in \mathcal{X}} f_t(x) \right)$, is impossible even with a restriction!

How do we measure how well we are doing?

- We compare ourselves with the ***best fixed strategy***: $x \in \mathcal{X}$.

Definition (Regret)

Given a sequence of functions $f_1, \dots, f_T$, the regret $\mathcal{R}(T)$ of the sequence $(x^1, \dots, x^T)$ is defined as

$$\mathcal{R}(T) := \sum_{t=1}^{T} f_t(x^t) - \underbrace{\min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x)}_{\text{cost of the best fixed-action}}. \tag{1}$$

Remarks:

- The concept of regret first appears in Hanan et. al [13].
- Other works contributing in the formalization are Blackwell et al. [7] and Vovk et al. [26].
- The *notion of regret* is a natural extension of *optimality* in (offline) convex optimization.

Online vs offline convex optimization

Goal of (online) convex optimization

Given a sequence of convex functions $f_1, \dots, f_T$, select a sequence $x^1, \dots, x^T \in \mathcal{X}$ (where $x^t \in \mathcal{X}$ is solely decided by $x^0, \nabla_1, \dots, \nabla_{t-1}$) with regret $\mathcal{R}(T) = o(T)$.

Remarks:

- If the regret of the sequence $x^1, \dots, x^T \in \mathcal{X}$ equals $\mathcal{R}(T) = o(T)$ then

$$\frac{1}{T}\left(\sum_{t=1}^{T} f_t(x^t) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x)\right) \leq \frac{\mathcal{R}(T)}{T} = \frac{o(T)}{T} \to 0.$$

- The *time-averaged cost of* $x^1, \dots, x^T$ approaches the *cost of best fixed action* $x^\star \in \mathcal{X}$*!*

Online-to-offline conversion (Convex)

- Let $\mathcal{R}(T)$ be the regret of sequence $x^1, \dots, x^T \in \mathcal{X}$ for the **constant** sequence of functions $f_1, \dots, f_T = f$. Then, by the convexity of f, we have

$$f\left(\frac{1}{T}\sum_{t=1}^{T} x_T\right) - \min_{x \in \mathcal{X}} f(x) \leq \frac{1}{T}\left(\sum_{t=1}^{T} f(x_t) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f(x)\right) \leq \frac{\mathcal{R}(T)}{T}.$$

- If $\mathcal{R}(T) = o(T)$, then $\lim_{t \to \infty} \mathcal{R}(T)/T = 0$.

Online convex optimization

Online learning algorithm (Hazan el al. [14])

An online learning algorithm $\mathcal{A}$ for an online convex optimization setting (with a feasibility set $\mathcal{X}$) outputs $x^t \in \mathcal{X}$ solely based on $(x^0, \nabla_1, \ldots, \nabla_{t-1})$, that is $x^t := \mathcal{A}\left(x^0, \nabla_1, \ldots, \nabla_{t-1}\right)$. Recall that $\nabla_t = \nabla_t f_t(x^t)$.

No-regret

An online learning algorithm $\mathcal{A}$ is called **no-regret** iff for any sequence of functions $f_1, \ldots, f_T$, $\mathcal{R}_{\mathcal{A}}(T) = o(T)$.

Brief history of no-regret algorithms:

- Hanan et. [13]: first *no-regret algorithm* with regret $\mathcal{O}\left(\sqrt{nT}\right)$ for $\mathcal{X} = \Delta_n$ and $f_t(x) := \langle c^t, x\rangle$.
- Littlestone et al. [19]: first $\mathcal{O}(\sqrt{T \log n})$-regret for $\mathcal{X} = \Delta_n$, $f_t(x) := \langle c^t, x\rangle$ and $c^t \in \{0,1\}^n$.
- Freund et al. [11]: The Hedge algorithm achieves $\mathcal{O}(\sqrt{T \log n})$ regret for $\mathcal{X} = \Delta_n$, $f_t(x) := \langle c^t, x\rangle$.
- Zinkevic et al. [27]: first $\mathcal{O}(\sqrt{T})$-regret for general convex set $\mathcal{X}$ and convex functions f_t.
- Abernethy et al. [3]: *Follow the Regularized Leader algorithm* for general convex sets and functions.

The expert problem

Online decision making

A *learner* needs to decide over n possible actions with **unknown and changing rewards** over T rounds.

The expert problem (Littlestone et al. [19])

At each round $t = 1, \dots, T$:

- A learner selects a prob. distribution $x^t \sim \{1, \dots, n\}$ over the n possible actions.
- An adversary selects a cost c_i^t for each action $i \in \{1, \dots, n\}$.
- The learner suffers a(n) (expected) cost $f_t(x^t) := \langle c^t, x^t \rangle$ and receives c^t as feedback ($c^t = \nabla f_t(x^t)$).

Remarks:

- Special case of OCO with linear functions $f_t(x^t) := \langle c^t, x^t \rangle$ and $\mathcal{X} = \Delta_n$.
- Suppose the learner selects $x^t \in \Delta_n$ according to a no-regret algorithm.
- Then its time-averaged cost is at most the time-averaged cost of best fixed action:

$$\frac{1}{T}\sum_{t=1}^{T}\langle c^t, x^t \rangle \leq \frac{1}{T}\min_{i\in[n]}\sum_{t=1}^{T} c_i^t + \underbrace{\frac{o(T)}{T}}_{\text{goes to } 0} !$$

Application of the expert problem: Online path selection (Awerbuch et al. [5])

Example (Going to Work)

- Every day $t \in \{1, \dots, T\}$, we go from home to work (and vice versa).
- There are multiple routes the travel time of which depends on unpredictable (weather, congestion) factors.
- How do we select our route every day?

Reduction to the expert problem

- Consider the expert problem by enumerating each possible route as a different action.
- Iterate $t \in \{1, \dots, T\}$:
  - Randomly select a route with probability distribution $x^t \in \Delta_n$.
  - Observe c_i^t as the *travel time* of the i-th route at day t observed after selection. Form a vector c^t.
  - Update $x^{t+1} \leftarrow \mathcal{A}(c^1, \dots, c^t)$ where $\mathcal{A}$ is a no-regret algorithm for the expert problem.
- If $\mathcal{A}$ is a no-regret algorithm for the expert problem, then the overall travel (over the T days) is approximately the travel of *the best fixed route*!

Old school spam filtering

Spam Filtering

Classify an e-mail as *spam* or *no-spam*.

Online spam filtering with linear filters (Hazan et al. [14])

- A dictionary $\mathrm{Dict}[\cdot]$ of length d containing all possible words.
- An arrived email is encoded as a $\{0,1\}$-vector $m_t \in \{0,1\}^d$ of length d depending on the contained words.
- We select $x^t \in [-1,1]^d$ and classify the email according to the *linear filter* $\hat{b}_t := \mathrm{sign}(m_t^\top x^t)$

$$\text{Decision at round } t \ := \begin{cases} \text{Spam} & \text{if } \hat{\beta}_t = -1 \\ \text{Inbox} & \text{if } \hat{\beta}_t = 1 \end{cases}$$

- The *true label* $b_t \in \{-1,1\}$ is revealed and we incur loss $(\hat{b}_t - b_t)^2$

Remarks:

- A no-regret algorithm select x^t with $\mathcal{X} = [-1,1]^d$ and $f_t(x) := \Big(\beta_t - \mathrm{sign}(\alpha^\top x)\Big)^2$.
- Then we can obtain comparable classification accuracy *with the best fixed linear filter*!

Old school spam filtering

Spam Filtering

Classify an e-mail as *spam* or *no-spam*.

Online spam filtering with linear filters (Hazan et al. [14])

- A dictionary $\mathrm{Dict}[\cdot]$ of length d containing all possible words.
- An arrived email is encoded as a $\{0,1\}$-vector $m_t \in \{0,1\}^d$ of length d depending on the contained words.
- We select $x^t \in [-1,1]^d$ and classify the email according to the *linear filter* $\hat{b}_t := \mathrm{sign}(m_t^\top x^t)$

$$\text{Decision at round } t \ := \begin{cases} \text{Spam} & \text{if } \hat{\beta}_t = -1 \\ \text{Inbox} & \text{if } \hat{\beta}_t = 1 \end{cases}$$

- The *true label* $b_t \in \{-1,1\}$ is revealed and we incur loss $(\hat{b}_t - b_t)^2$

Remarks:

- If a no-regret algorithm select x^t with $\mathcal{X} = [-1,1]^d$ and $f_t(x) := \left(\beta_t - \mathrm{sign}(\alpha^\top x)\right)^2$.
- Then we can obtain comparable classification accuracy *with the best fixed linear filter*!
- Unfortunately, the problem is not OCO so this proposition serves only as motivation.

Portfolio management

Portfolio Management

- We start with a total capital of C_0 dollars.
- Each day $t \in \{1, \dots, T\}$, we want to invest our capital in n possible assets so as to maximize our profit.
- The *return* $r_i^t > 0$ of asset i is the *price ratio* of asset i at the beginning and at the end of day t.
- The choice of not investing can be encoded with a special asset 0 with $r_0^t = 1$.

Universal portfolio problem (Cover et al. [10], Kalai et al. [15], Tsai et al. [24])

- At the beginning of each day $t \in \{1, \dots, T\}$: the decision maker splits its current capital over the n possible assets according to the distribution $x_t \in \Delta_n$.
- At the end of each day $t \in \{1, \dots, T\}$: the decision maker's capital becomes $C_{t+1} := C_t \cdot \langle r_t, x^t \rangle$.
- The decision maker learns the *return vector* $r^t \in \mathbb{R}_+^n$ and updates $x^{t+1} \in \Delta_n$ so as to minimize

$$\underbrace{\max_{x \in \Delta_n} \sum_{t=1}^{T} \log\left(\langle r_t, x \rangle\right) - \sum_{t=1}^{T} \log\left(\langle r_t, x_t \rangle\right).}_{\text{The constant rebalancing portfolio problem}}$$

Portfolio management II

Connection with online convex optimization (OCO)

Universal portfolio problem fits into the OCO setting with $\mathcal{X} = \Delta_n$ and $f_t(x) := -\log(\langle r_t, x \rangle)$.

Example

Consider initial capital $C_0 = 1$, assets A, B with return sequence $(r_A^1 = 100, r_B^1 = 1)$, $(r_A^2 = 0.01, r_B^2 = 1.5)$.

- Investing always in A produces 1.
- Investing always in B produces 1.5.
- Splitting the capital at each round equally between A, B $(x^1 = x^2 = (0.5, 0.5))$ produces $\simeq 36.4$.
- Best fixed splitting uses $x = \left(\frac{14701}{29502}, \frac{14801}{29502}\right)$, resulting in $\simeq 36.41$.

Remarks:

- In the universal portfolio problem, the *best fixed investment strategy* is not necessarily "pure."
- The universal portfolio problem is related to quantum tomography [25].
- No regret algorithms exist that leverage the self-concordance of f_t's [25].

Recall: The expert problem

The expert problem (Littlestone et al. [19])

At each round $t = 1, \dots, T$, we have

- A learner selects a probability distribution $x^t \in \Delta_n$ over n possible actions.
- An adversary selects a cost $c_i^t \in [-1, 1]$ for each action $i \in \{1, \dots, n\}$.
- The learner suffers an expected cost $\langle c^t, x^t \rangle$ and receives $c^t \in [-1, 1]^n$ as feedback.

Remark: ○ The expert problem is a special case of OCO with $\mathcal{X} = \Delta_n$ and $f_t(x) := \langle c^t, x \rangle$.

The Hedge algorithm

The Hedge algorithm (Freund et al. [11])

- Initialize expert weights $w^1 \leftarrow (1, \dots, 1)$
- For each round $t = 1, \dots, T$
  - The *learner* selects a probability distribution $x^t \in \Delta_n$ as follows,

$$x_i^t = \frac{w_i^t}{\sum_{j=1}^n w_j^t} \text{ for each action } i \in \{1, \dots, n\}.$$

  - The adversary selects a cost $c_i^t \in [-1, 1]$ for each action $i \in \{1, \dots, n\}$.
  - The *learner* suffers expected $\langle c^t, x^t \rangle$ and receives $c^t \in [-1, 1]^n$ as feedback.
  - The *learner* updates the weights as follows,

$$w_i^{t+1} := w_i^t e^{-\gamma c_i^t}$$

    where $\gamma > 0$ is the learning rate.

Remark:

- Hedge is closely connected with two methods:
  - the *dual averaging method* with entropic regularization of Nesterov et al. [20]
  - the entropic mirror descent Beck and Teboulle [6]
- These methods coincide when the objective is linear and the constraint is the simplex.

The Hedge algorithm

Remark: ◦ The Hedge algorithm admits regret $\mathcal{R}_{\mathrm{Hedge}}(T) = \mathcal{O}\left(\sqrt{T \log n}\right)$.

Theorem (Freund et al. [11])

The Hedge algorithm with the step-size $\gamma := \sqrt{\log n / T}$ *admits regret* $\mathcal{R}_{\mathrm{Hedge}}(T) = \mathcal{O}\left(\sqrt{T \log n}\right)$. *More precisely, for any cost-vector sequence* $c^1, \ldots, c^T \in [-1, 1]^n$, *it holds that*

$$\sum_{t=1}^{T} \langle c^t, x^t \rangle \leq \min_{x \in \Delta_n} \sum_{t=1}^{T} \langle c^t, x \rangle + \mathcal{O}\left(\sqrt{T \log n}\right).$$

The Hedge algorithm: Proof I

Proof.

Let $\Phi(t) = \sum_{i=1}^{n} w_i^t$ meaning that $\Phi(1) = n$. Then, it follows that

$$
\begin{aligned}
\Phi(t+1) &= \sum_{i=1}^{n} w_i^{t+1} = \sum_{i=1}^{n} w_i^t e^{-\gamma c_i^t} \\
&= \Phi(t) \sum_{i=1}^{n} x_i^t e^{-\gamma c_i^t} \\
&\leq \Phi(t) \sum_{i=1}^{n} x_i^t \left(1 - \gamma c_i^t x_i^t + \gamma^2 (c_i^t)^2 x_i^t\right) && (e^{-x} \leq 1 - x + x^2) \\
&= \Phi(t) \left(1 - \gamma \langle c^t, x^t \rangle + \gamma^2 \langle c^{2t}, x^t \rangle\right) && c^{2t} = \left((c_1^t)^2, \ldots, (c_n^t)^2\right) \\
&\leq \Phi(t) e^{-\gamma \langle c^t, x^t \rangle + \gamma^2 \langle c^{2t}, x^t \rangle} && (1 - x \leq e^{-x}) \\
&\leq \Phi(1) e^{-\gamma \sum_{t=1}^{T} \langle c^t, x^t \rangle + \gamma^2 \sum_{t=1}^{T} \langle c^{2t}, x^t \rangle} && \text{(recursion)} \\
&\leq n e^{-\gamma \sum_{t=1}^{T} \langle c^t, x^t \rangle + \gamma^2 T} && (c^{2t} \in [0,1]^n)
\end{aligned}
$$

The Hedge algorithm: Proof II

Proof (Cont.)

Let $i^\star \in \{1, \dots, n\}$ be the optimal fixed action, $i^\star := \operatorname{argmin}_{i \in \{1,\dots,n\}} \sum_{t=1}^T c_i^t$. Then,

$$e^{-\gamma \sum_{t=1}^T c_{i^\star}^t} = w_{i^\star}^{T+1} \leq \sum_{i=1}^n w_i^{T+1} \quad = \quad \Phi(T+1) \leq n e^{-\gamma \sum_{t=1}^T \langle c^t, x^t \rangle + \gamma^2 T}$$

As a result,

$$\sum_{t=1}^T \langle c^t, x^t \rangle \leq \min_{i \in \{1,\dots,n\}} \sum_{t=1}^T c_i^t + \frac{\log n}{\gamma} + \gamma T.$$

The proof is concluded by selecting $\gamma := \sqrt{\log n / T}$.

Remarks:

- The step-size can be chosen in an iteration dependent way.
- See the entropic mirror descent derivation in [6].

Online projected gradient descent

Remarks:

- Hedge provides no-regret guarantees for the OCO setting with $\mathcal{X} = \Delta_n$ and $f_t(x) := \langle c^t, x \rangle$.
- Online projected gradient descent provides no-regret guarantees for projectable convex sets $\mathcal{X}$.

Online projected gradient descent (Zinkevic et al. [27])

- At each round $t = 1, \dots, T$
  - The *learner* selects an $x^t \in \mathcal{X}$.
  - The *adversary* selects a convex function $f_t \in \mathcal{F} : \mathcal{X} \mapsto \mathbb{R}$.
  - The *learner* suffers cost $f_t(x^t)$ and receives $\nabla_t := \nabla f_t(x^t)$ as feedback
  - The learner updates $x^{t+1} \in \mathcal{X}$ as follows,

$$x^{t+1} \leftarrow \Pi_{\mathcal{X}} \left(x^t - \gamma \nabla_t \right) \quad \text{(Online GD)}$$

  where $\gamma > 0$ is the learning rate.

Online gradient descent: A basic proof - I

Theorem (Zinkevic et al. [27])

For any sequence of convex differentiable functions $f_1, \dots, f_T$ satisfying $\max_{x \in \mathcal{X}} \|\nabla f_t(x)\| \leq G$ (i.e., $\mathcal{F}$), online projected gradient descent with step-size $\gamma = |\mathcal{X}|/G\sqrt{T}$, admits $\mathcal{O}\left(\sqrt{G|\mathcal{X}|T}\right)$ regret. More precisely,

$$\sum_{t=1}^{T} f_t(x^t) \leq \min_{x^\star \in \mathcal{X}} \sum_{t=1}^{T} f_t(x^\star) + \mathcal{O}(G|\mathcal{X}|\sqrt{T}).$$

Proof.

$$\begin{aligned} \|x^{t+1} - x^\star\|^2 &= \|\Pi_{\mathcal{X}}\left(x^t - \gamma \nabla_t\right) - x^\star\|^2 \\ &\leq \|x^t - \gamma \nabla_t - x^\star\|^2 \qquad \text{(non-expansiveness of convex projections)} \\ &= \|x^t - x^\star\|^2 - 2\gamma \nabla_t^\top (x^t - x^\star) + \gamma^2 \|\nabla_t\|^2 \end{aligned}$$

Thus,

$$\nabla_t^\top (x^t - x^\star) \leq \frac{\|x^t - x^\star\|^2 - \|x^{t+1} - x^\star\|^2}{2\gamma} + \frac{\gamma}{2}\|\nabla_t\|^2$$

□

Online gradient descent: A basic proof - II

Proof (Cont.)

As a result,

$$\begin{aligned}
\sum_{t=1}^{T} f_t(x_t) - f_t(x^\star) &\leq \sum_{t=1}^{T} \nabla_t^\top (x^t - x^\star) \\
&\leq \sum_{t=1}^{T} \frac{\|x^t - x^\star\|^2 - \|x^{t+1} - x^\star\|^2}{2\gamma} + \frac{\gamma}{2} \sum_{t=1}^{T} \|\nabla_t\|^2 \\
&\leq \frac{\|x^1 - x^\star\|^2}{2\gamma} + \frac{\gamma}{2} \sum_{t=1}^{T} \|\nabla_t\|^2 \\
&\leq \frac{|\mathcal{X}|^2}{2\gamma} + \frac{\gamma G^2 T}{2} \\
&= |\mathcal{X}| G \sqrt{T} \quad \text{for} \quad \gamma := |\mathcal{X}| / G\sqrt{T}.
\end{aligned}$$

Let's take a breather

Algorithm $\mathcal{A}$	Regret $\mathcal{R}$	Function class $\mathcal{F}$	Feasibility set $\mathcal{X}$
Hedge	$\mathcal{O}\left(\sqrt{\log n T}\right)$	Linear functions	n-dimensional simplex Δ_n
Online gradient descent (OGD)	$\mathcal{O}\left(G\lvert\mathcal{X}\rvert\sqrt{T}\right)$	G-Lipschitz	General convex set $\mathcal{X}$

Remarks:

- Hedge and OGD look like ad-hoc approaches transferring algorithms from the offline setting
- In the sequel, we build up a more structural approach specifically for the online setting
  - The follow the regularized leader (FTRL) class of algorithms [3, 23, 17]

A first (naive) attempt

- How to pick the next $x^t \in \mathcal{X}$ given $f_1, ..., f_t$?
- A naive first attempt: Follow the leader (FTL) Kalai et al. [16], which picks the *best strategy so far*:

$$x^t = \arg\min_{x \in \mathcal{X}} \sum_{\tau=1}^{t-1} f_\tau(x) \qquad \text{(FTL)}$$

A first (naive) attempt

- How to pick the next $x^t \in \mathcal{X}$ given $f_1, ..., f_t$?
- A naive first attempt: Follow the leader (FTL) Kalai et al. [16], which picks the *best strategy so far*:

$$x^t = \arg\min_{x \in \mathcal{X}} \sum_{\tau=1}^{t-1} f_\tau(x) \qquad \text{(FTL)}$$

- Unfortunately a simple adversarial strategy exists.

Example (Adversarial strategy against FTL (Shalev-Shwartz [22, Ex. 2.2]))

Consider $\mathcal{X} = \{-1, 1\}$ and the environment picking

t	1	2	3	4	$\cdots$
$f_t(x)$	$\frac{1}{2}x$	$-x$	x	$-x$	$\cdots$
$\sum_{\tau=1}^{t-1} f_\tau(x)$	-	$\frac{1}{2}x$	$-\frac{1}{2}x$	$\frac{1}{2}x$	$\cdots$

Remark:

- (FTL) picks smallest $x^t = -1$ when minimizer of $f_t(\cdot)$ is largest x (and vice versa).
- So (FTL) achieves *maximal* regret $f_{t+1}(x^{t+1})$.
- Can we do better?

What if we could cheat?

- Be the leader (BTL): Imagine we could cheat and use x^{t+1} at time t while incurring the cost.
- Now, we incur $f_t(x^{t+1})$ instead of $f_t(x^t)$ in the regret analysis.

Lemma (Regret of BTL)

Let BTL generate the sequence $(x^1, \dots, x^{T+1})$ according to FTL but play x^{t+1} at time t. Then, BTL admits non-positive regret:

$$\mathcal{R}_{\mathrm{BTL}}(T) := \sum_{t=1}^{T} f_t(x^{t+1}) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x) \leq 0.$$

Proof.

$$\begin{aligned}
\mathcal{R}_{\mathrm{BTL}}(T) &= \sum_{t=1}^{T} f_t(x^{t+1}) - \min_x \sum_{t=1}^{T} f_t(x) && && \text{(2a)}\\
&= \sum_{t=1}^{T} f_t(x^{t+1}) - \sum_{t=1}^{T} f_t(x^{T+1}) && \text{(by def of } x^t\text{)} && \text{(2b)}\\
&= \sum_{t=1}^{T-1} f_t(x^{t+1}) - \sum_{t=1}^{T-1} f_t(x^{T+1}) && \text{(last terms are equal)} && \text{(2c)}\\
&\leq \sum_{t=1}^{T-1} f_t(x^{t+1}) - \sum_{t=1}^{T-1} f_t(x^t) && \text{(since } x^T \text{ is the actual minimum)} && \text{(2d)}\\
&\leq f_1(x^2) - f_1(x^3) && \text{(recurse until } T - i = 1 \text{ in (2c))} && \text{(2e)}\\
&\leq 0 && \text{(since } f_1(x^2) \text{ is the minimum.)} && \text{(2f)}
\end{aligned}$$

Implications of BLT's non-positive regret

- BTL's non-positive regret provides an interesting insight for FTL:

Insight

The regret of the sequence $(x^1, \dots, x^T)$, as generated by (FTL), is no worse than

$$\mathcal{R}_{\mathrm{FTL}}(T) = \sum_{t=1}^{T} f_t(x^t) - \min_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x) \leq \sum_{t=1}^{T} (f_t(x^t) - f_t(x^{t+1})). \tag{3}$$

Remarks:

- Follows from $\mathcal{R}_{\mathrm{FTL}}(T) - 0 \leq \mathcal{R}_{\mathrm{FTL}}(T) - \mathcal{R}_{\mathrm{BTL}}(T)$ which is immediate from BTL.
- Consequently, $\mathcal{R}_{\mathrm{FTL}}(T)$ is no worse than the difference between $f_t(x^t)$ and $f_t(x^{t+1})$

Observations:

- *Intuitively, we just need to ensure x^t does not change too much.*
- *We should regularize/stabilize!*

Follow the *regularized* leader (FTRL)

Follow the regularized leader (Abernethy et al. [4])

- For each round $t = 1, \dots, T$
  - The *learner* selects $x^t \in \mathcal{X}$ using

$$x^t = \arg\min_{x \in \mathcal{X}} \sum_{\tau=1}^{t-1} f_\tau(x) + \tfrac{1}{\gamma} h(x) \qquad \text{(FTRL)}$$

    where $\gamma > 0$ will be (!) the learning rate and $h : \mathcal{X} \to \mathbb{R}$ is a strongly-convex regularizer in some norm $\|\cdot\|$.
  - The *adversary* selects a function $f_t(\cdot)$ where $f_t : \mathcal{X} \mapsto \mathbb{R}$.
  - The learner suffers $f_t(x^t)$ and gets access to $f(\cdot)$.

Remarks:

- We modified the selection of x^t.
- The regularizer ensures iterates do not move too much and forces uniqueness of solution.

Historical notes:

- Regularization was studied in online learning by Grove et al. [12] and Kivinen et al. [18]
- Follow the leader (FTL) was coined by the influencial paper Kalai et al. [16]
- FTRL Introduced in Shalev-Shwartz [21] and Abernethy et al. [4] almost simultaneously.

Stability due to regularization

Lemma (Stability)

The sequence $(x^1, \dots, x^T)$ *generated by FTRL satisfies*

$$f_t(x^t) - f_t(x^{t+1}) \leq \gamma \|\nabla f_t(x^t)\|_*^2, \tag{4}$$

where $\|\cdot\|_*$ *is the dual norm of* $\|\cdot\|$ *defined as* $\|x\|_* = \sup_{\|y\|\leq 1} \langle x, y \rangle$.

Proof.

Let us define the function we minimizes in the decision selection

$$F_t(x) = \sum_{\tau=1}^{t} f_\tau(x) + \tfrac{1}{\gamma} h(x). \tag{5}$$

By strong convexity of h and convexity of f_τ we have that

$$F_{t-1}(x^t) - F_{t-1}(x^{t+1}) \leq \langle \nabla F_{t-1}(x^t), x^t - x^{t+1} \rangle - \tfrac{1}{2\gamma} \|x^t - x^{t+1}\|^2 \quad \text{for any } x^t, x^{t+1} \in \mathcal{X}. \tag{6}$$

This will ultimately let us bound $f_t(x^t) - f_t(x^{t+1})$. (cont.) □

Stability

Proof (Cont.)

FTRL defines $x^t = \arg\min_x F_{t-1}(x)$ so $F_{t-1}(x^t)$ is the optimum and first order characterization[1] becomes

$$\langle \nabla F_{t-1}(x^t), x^t - x^{t+1} \rangle \leq 0. \tag{7}$$

So a weaker bound of (6) is

$$\begin{aligned} F_{t-1}(x^t) - F_{t-1}(x^{t+1}) &\leq -\tfrac{1}{2\gamma}\|x^t - x^{t+1}\|^2 \quad \text{and} \\ F_t(x^{t+1}) - F_t(x^t) &\leq -\tfrac{1}{2\gamma}\|x^t - x^{t+1}\|^2. \end{aligned} \tag{8}$$

The second line simply applies the same reasoning. We can now sum the two lines and expand the definition of F_i. All terms will cancel out except the last term in $F_t(x^{t+1})$ and $F_t(x^t)$ so we get

$$f_t(x^{t+1}) - f_t(x^t) \leq -\tfrac{1}{y}\|x^t - x^{t+1}\|^2 \;\Leftrightarrow\; \|x^t - x^{t+1}\|^2 \leq \gamma(f_t(x^t) - f_t(x^{t+1})). \tag{9}$$

(cont.)

[1] First order characterization of convexity says $f(y) - f(x) \leq \nabla f(y)^\top (y - x)$. So when $f(y)$ is minimum we have $\nabla f(y)^\top (y - x) \geq 0$.

Stability

Proof (Cont.)

Now we have the tools to bound the change.[2]

$$\begin{aligned} f_t(x^t) - f_t(x^{t+1}) &\leq \langle \nabla f_t(x^t), x^t - x^{t+1} \rangle && \text{(Convexity)} \\ &\leq \|\nabla f_t(x^t)\|_* \|x^t - x^{t+1}\| && \text{(Hölder's ineq.)} \\ &\leq \|\nabla f_t(x^t)\|_* \sqrt{\gamma} \sqrt{f_t(x^t) - f_t(x^{t+1})} && \text{from (9)} \end{aligned} \tag{10}$$

Solving for $f_t(x^t) - f_t(x^{t+1})$, we get

$$f_t(x^t) - f_t(x^{t+1}) \leq \gamma \|\nabla_t f(x^t)\|_*^2. \tag{11}$$

[2] Hölder's inequality: $\langle x, y \rangle \leq \|x\| \|y\|_*$.

Regret of FTRL

- Equipped with stability and the BTL lemma we almost directly obtain the regret bound for FTRL.

Theorem (Regret of FTRL ([21, 4]))

The sequence $(x^1, \dots, x^T)$ *generated by* (FTRL) *satisfies*

$$\mathcal{R}_{\mathrm{FTRL}}(T) \leq \frac{R_h}{\gamma} + \gamma \sum_{t=1}^{T} \|\nabla f_t(x^t)\|_*^2, \tag{12}$$

where $R_h := \max_x h(x) - \min_x h(x)$.

Remark:

- The particular presentation of the proof below is due to Luo [1].

Proof.

By defining $f_0(x) = \frac{1}{\gamma} h(x)$ we can write the regularized selection as

$$x^t = \arg\min_{x \in \mathcal{X}} \sum_{\tau=1}^{t-1} f_\tau(x) + \frac{1}{\gamma} h(x) = \arg\min_{x \in \mathcal{X}} \sum_{\tau=0}^{t-1} f_\tau(x). \tag{13}$$

(cont.) □

Regret of FTRL

Proof (Cont.)

First note that by the BTL lemma we have[3]

$$\sum_{t=0}^{T} f_t(x^\star) \geq \sum_{t=0}^{T} f_t(x^{t+1}), \tag{14}$$

where $x^\star = \arg\min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x)$ is the best fixed decision. Now we can bound the FTRL regret as follows[4]

$$\begin{aligned}
\mathcal{R}_T^{\mathrm{FTRL}} &= \sum_{t=1}^{T} f_t(x^t) - \sum_{t=1}^{T} f_t(x^\star) \\
&= \sum_{t=1}^{T} f_t(x^t) - \sum_{t=1}^{t} f_t(x^\star) + f_0(x^\star) \\
&\leq \sum_{t=1}^{T} f_t(x^t) - \sum_{t=0}^{T} f_t(x^{t+1}) + f_0(x^\star) && \text{(using BTL lemma)} \\
&\leq \gamma \sum_{t=1}^{T} \|\nabla_t f(x^t)\|_*^2 + f_0(x^\star) - f_0(x^1) && \text{(using stability)} \\
&\leq \gamma \sum_{t=1}^{T} \|\nabla_t f(x^t)\|_*^2 + \underbrace{\tfrac{1}{\gamma}(\max_x h(x) - \min_{x\in\mathcal{X}} h(x))}_{R_h}.
\end{aligned} \tag{15}$$

[3] Note that we additionally use $f_0(x^\star) \geq f_0(x^1)$ since BTL only applies to $t = 1, \ldots, T$.

[4] We want to use stability so we need to move from $x^\star$ to x^{t+1}. We do this by getting it on a form for which we can apply the BTL lemma.

Regret of FTRL

Corollary

Further, if $f_t(x)$ is G-Lipschitz and we choose a learning rate of $\gamma = \sqrt{\frac{R_h}{TG^2}}$, then we have

$$\mathcal{R}_{\mathrm{FTRL}}(T) = \mathcal{O}(G\sqrt{TR_h}). \tag{16}$$

Remark:

- G-Lipschitz assumption ensures that the gradient is bounded: $\|\nabla f_t(x^t)\|_* \leq G$.
- The "optimal" γ is found by simply optimizing the regret bound which is of the form

$$\arg\min_x \{ax + \tfrac{b}{x}\} = \sqrt{\tfrac{b}{a}}. \tag{17}$$

- FTRL has sublinear regret.

Linear losses are sufficient for OCO in general

- Under convexity it suffices to have access to gradient through $\langle \nabla f_t(x^t), \cdot \rangle$ instead of the whole function $f_t(\cdot)$:

$$\mathcal{R}_{\mathrm{FTRL}}(T) = \max_{x \in \mathcal{X}} \sum_{t=1}^{T} f_t(x^t) - f_t(x) \leq \max_{x \in \mathcal{X}} \sum_{t=1}^{T} \langle \nabla f_t(x^t), x^t - x \rangle \quad \textit{by convexity.} \tag{18}$$

- Indeed, the regret for $f_t(x)$ is bounded by the regret for another problem based on $f'_t(x) = \langle \nabla f_t(x^t), x \rangle$.

FTRL with only gradients

- For each round $t = 1, \ldots, T$
  - The *learner* selects $x_t \in \mathcal{X}$ using

$$x^{t+1} = \arg\min_{x \in \mathcal{X}} \left\langle \sum_{\tau=1}^{t} \nabla f_\tau(x^\tau), x \right\rangle + \tfrac{1}{\gamma} h(x) \qquad \text{(FTRL on gradients)}$$

    where $\gamma > 0$ is the learning rate and $h : \mathcal{X} \to \mathbb{R}$ is a strongly-convex regularizer in some norm $\| \cdot \|$.
  - The *adversary* selects a $f_t(\cdot)$.
  - The learner suffers $f_t(x^t)$ and observes $\nabla f_t(x^t)$.

Remark:

- FTRL with linear losses is closely connected to two offline algorithms:
  - equivalent to the *dual averaging method* of Nesterov et al. [20]
  - coincides with the entropic mirror descent Beck and Teboulle [6] under simplex constraints

Summary of no-regret algorithms

Algorithm $\mathcal{A}$	Regret $\mathcal{R}$	Function class $\mathcal{F}$	Feasibility set $\mathcal{X}$
Hedge	$\mathcal{O}\left(\sqrt{\log n T}\right)$	Linear functions	n-dimensional simplex Δ_n
Online gradient descent (OGD)	$\mathcal{O}\left(G\lvert\mathcal{X}\rvert\sqrt{T}\right)$	G-Lipschitz	General convex set $\mathcal{X}$
Follow the regularized leader (FTRL)	$\mathcal{O}\left(GR_h\sqrt{T}\right)$	G-Lipschitz	General convex set $\mathcal{X}$

Remarks:

- By all general convex sets, we mean all *projectable* general convex sets. Why?
- All algorithms that we saw so far admit $\mathcal{O}(\sqrt{T})$ regret. Can we do better?

Lower bounds

Answer: ○ Unfortunately no!

Theorem (Lower bound (Abernethy et al. [2, Lm. 8]))

Let $\mathcal{X} = \mathbb{B}(0,1)$ (n-dimensional unit ball centered at $(0,\dots,0)$. Then any online learning algorithm $\mathcal{A}$ admits regret greater than $\sqrt{T}$.

Proof

At each round t, the adversary selects c^t such that the following hold:

- $\langle c^t, x_t, \rangle = 0$,
- $\langle c^t, \sum_{s=1}^{t-1} c_s \rangle = 0$,
- $\|c^t\| = 1$.

By the construction $\sum_{t=1}^{T} \langle c^t, x_t, \rangle = 0$. Consider $x^\star = -\sum_{t=1}^{T} c^t / \| \sum_{t=1}^{T} c^t \|$, then it holds that

$$\sum_{t=1}^{T} \langle c^t, x^\star \rangle = -\| \sum_{t=1}^{T} c^t \|.$$

(cont.)

Lower bounds

Proof (Cont.)

Let us try to find how big $\| \sum_{t=1}^{T} c^t \|$ can be

$$\begin{aligned} \| \sum_{s=1}^{t} c_s \|^2 &= \| \sum_{s=1}^{t-1} c_s + c^t \|^2 \\ &= \| \sum_{s=1}^{t-1} c_s \|^2 + \underbrace{2 \langle c^t, \sum_{s=1}^{t-1} c_s \rangle}_{0} + \| c^t \|^2 \\ &= \| \sum_{s=1}^{t-1} c_s \|^2 + 1 \end{aligned}$$

Thus $\| \sum_{t=1}^{T} c^t \| = \sqrt{T}$. As a result,

$$\sum_{t=1}^{T} c^t (x_t - x^\star) = -\sqrt{T}.$$

A tighter lower bound for the expert problem

- For the expert problem (simplex constraints) we can characterize dependency on action cardinality n.
- We will show that our upper bound can at best be improved by a constant factor.

Theorem (Cesa-Bianchi et al. [9, Thm. 3.7])

For any online learning algorithm $\mathcal{A}$ for the expert problem $\mathcal{X} = \Delta_n$, the regret satisfies the following

$$\mathcal{R}_{\mathcal{A}}(T) \geq \frac{\sqrt{T \ln n}}{\sqrt{2}}. \tag{19}$$

Remarks:

- A deterministic construction might be difficult to find.
- **Trick** instead uses that any probabilistic construction will lower bound the supremum

$$\sup_{z \in \mathcal{Z}} f(z) \geq \mathbb{E}_z[f(z)]. \tag{20}$$

- The proof presentation in the sequel is from Haipeng Luo CSCI 699 lecture notes.

A tighter lower bound for the expert problem

Proof.

Specifically, if we let P be uniform over $\{0,1\}$, then the following holds

$$\begin{aligned}
\max_{c^1,\dots,c^T} \mathcal{R}_{\mathcal{A}}(T) &\geq \mathbb{E}_{c^1,\dots,c^T \overset{\text{iid}}{\sim} \mathsf{P}}[\mathcal{R}_{\mathcal{A}}(T)] \\
&= \textstyle\sum_{t=1}^T \mathbb{E}_{c^1,\dots,c^{t-1}} \mathbb{E}_{c^t}[\langle p^t, c^t\rangle \mid c^{t-1},\dots,c^1] - \mathbb{E}_{c^1,\dots,c^T}[\min_{i\in[N]} \textstyle\sum_{t=1}^T c_i^t] \\
&= \textstyle\sum_{t=1}^T \mathbb{E}_{c^1,\dots,c^{t-1}} \langle p^t, \mathbb{E}_{c^t}[c^t \mid c^{t-1},\dots,c^1]\rangle - \mathbb{E}_{c^1,\dots,c^T}[\min_{i\in[N]} \textstyle\sum_{t=1}^T c_i^t] \\
&= T/2 - \mathbb{E}_{c^1,\dots,c^T}[\min_{i\in[N]} \textstyle\sum_{t=1}^T c_i^t] \qquad (21) \\
&= \mathbb{E}_{c^1,\dots,c^T}[\max_{i\in[N]} \textstyle\sum_{t=1}^T (\frac{1}{2} - c_i^t)] \\
&= \frac{1}{2}\mathbb{E}_{u^1,\dots,u^T}[\max_{i\in[N]} \textstyle\sum_{t=1}^T u_i^t],
\end{aligned}$$

where u^t are Rademacher random variables. It is then not difficult to show the following (see e.g. [9, Lemma A.11 and A.12] and [9, Thm 3.7])

$$\lim_{T\to\infty} \lim_{N\to\infty} \mathbb{E}_{u^1,\dots,u^T}[\max_{i\in[N]} \textstyle\sum_{t=1}^T u_i^t] = \sqrt{2T\ln n}. \qquad (22)$$

□

Application of online learning: Obtaining statistical guarantees

- A no-regret algorithm enjoys statistical guarantees in the offline setting.
- Suppose we want to minimize the true risk under some distribution P:

$$\min_{x \in \mathcal{X}} \mathbb{E}_{z \sim \mathrm{P}}[\ell(x, z)]. \tag{23}$$

Meta-algorithm (Online to batch conversion)

- Run the online learning algorithm on $f_t(x) = \ell(x, z_t)$ for $t = 1...T$ where $z_t \sim \mathrm{P}$.
- Use the average over all actions as the prediction, i.e., $\hat{x}^T = \frac{1}{T} \sum_{t=1}^{T} x_t$.

Remark:
- Notice that each data point z_t is only seen once.

Application of online learning: Obtaining statistical guarantees

Theorem (Online to batch conversion (Cesa-Bianchi et al. [8]))

If the loss $x \to \mathbb{E}_{z\sim P}\ell(x,z)$ is convex then the true risk can be bounded with probability at least $1-\delta$ as follows

$$\mathbb{E}_{z\sim P}[\ell(\hat{x}^T, z)] \leq \mathbb{E}_{z\sim P}[\ell(x^\star, z)] + \frac{\mathcal{R}_{\mathcal{A}}(T)}{T} + 2\sqrt{\frac{2\ln(2/\delta)}{T}}, \tag{24}$$

where $\mathcal{R}_{\mathcal{A}}(T)$ is the regret of the online learning algorithm $\mathcal{A}$ after T rounds.

Proof.

The claim follows directly from application of Jensen's inequality and Azuma's inequality. □

- What convergence rate can we achieve with online learning for classic iid. statistical learning problems?

Answer:

- For FTRL the regret bound is $\mathcal{R}_T = \mathcal{O}(G\sqrt{DT})$ when $f_t(x)$ is G-Lipschitz.
- So the convergence rate becomes $\mathcal{O}(\frac{1}{\sqrt{T}})$.

Application of online learning: Approximate Nash equilibrium in zero-sum

- Consider the following problem

$$\min_{x\in\mathcal{X}} \max_{y\in\mathcal{Y}} f(x,y), \tag{25}$$

where $f(\cdot, y)$ is convex and $f(x,\cdot)$ is concave for all x, y.

- Assume we run two no-regret algorithms, i.e., for any x and y,

$$\begin{aligned} \mathcal{R}_{\mathrm{y}}(T) &\leq \textstyle\sum_{i=1}^{T} f(x^t, y) - \sum_{i=1}^{T} f(x^t, y^t), \\ \mathcal{R}_{\mathrm{x}}(T) &\leq \textstyle\sum_{i=1}^{T} f(x^t, y^t) - \sum_{i=1}^{T} f(x, y^t). \end{aligned} \tag{26}$$

Theorem (Approximate Nash equilibrium)

Assume $f : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}$ is convex-concave. Consider the sequence $\{(x^t, y^t)\}_{t=1}^{T}$ generated by two no-regret algorithm executed in tandem. Then $\hat{x}^T = \frac{1}{T}\sum_{t=1}^{T} x^t$ and $\bar{y}^T = \frac{1}{T}\sum_{t=1}^{T} y^t$ satisfies,

$$f(\hat{x}^T, y) - \varepsilon_T \leq f(x^\star, y^\star) \leq f(x, \bar{y}^T) + \varepsilon_T, \tag{27}$$

where $\varepsilon_T := \frac{1}{T}(\mathcal{R}_{\mathrm{y}}(T) + \mathcal{R}_{\mathrm{x}}(T))$ and $(x^\star, y^\star)$ is a Nash equilibrium.

Remark:

- Consequently, the *average* iterate of a no-regret algorithm converges as $\varepsilon_T = \mathcal{O}(1/\sqrt{T})$.

Application of online learning: Approximate Nash equilibrium in zero-sum

Proof.

Using the no-regret property,

$$\begin{aligned} f(\hat{x}^T, y) &\leq \frac{1}{T}\sum_{i=1}^T f(x^t, y) \leq \max_y \tfrac{1}{T}\sum_{i=1}^T f(x^t, y) \leq \tfrac{1}{T}\sum_{i=1}^T f(x^t, y^t) + \tfrac{1}{T}\mathcal{R}_{\mathrm{y}}(T) \\ f(x, \bar{y}^T) &\geq \frac{1}{T}\sum_{i=1}^T f(x, y^t) \geq \min_x \tfrac{1}{T}\sum_{i=1}^T f(x, y^t) \geq \tfrac{1}{T}\sum_{i=1}^T f(x^t, y^t) - \tfrac{1}{T}\mathcal{R}_{\mathrm{x}}(T), \end{aligned} \tag{28}$$

Subtracting the two equations,

$$f(\hat{x}^T, y) - f(x, \bar{y}^T) \leq \frac{1}{T}(\mathcal{R}_{\mathrm{y}}(T) + \mathcal{R}_{\mathrm{x}}(T)) =: \varepsilon_T. \tag{29}$$

We wish to relate to the Nash equilibrium $(x^\star, y^\star)$, defines as $f(x^\star, y) \leq f(x^\star, y^\star) \leq f(x, y^\star)$ for all x, y. First by picking $x = x^\star$ in (29) and second by property of a Nash equilibrium we get,

$$f(\hat{x}^T, y) - \varepsilon_T \leq f(x, \bar{y}^T) = f(x^\star, \bar{y}^T) \leq f(x^\star, y^\star). \tag{30}$$

A similar argument applies to the y-player and we conclude that $(\bar{x}, \bar{y})$ is an ε_T-approximate Nash equilibrium, i.e.,

$$f(\hat{x}^T, y) - \varepsilon_T \leq f(x^\star, y^\star) \leq f(x, \bar{y}^T) + \varepsilon_T. \tag{31}$$

□

Wrap-up

- We have seen that $\mathcal{O}(\sqrt{T})$ is both an upper and lower bound on the regret.
- In the offline setting this gives a $\mathcal{O}(1/\sqrt{T})$ rate for convex-concave minimax problems.
- Next week will see how we can improve this to $\mathcal{O}(1/T)$ in the offline setting!
- See you next week!

References I

[1] CSCI 699: Introduction to Online Learning, 2019.
(Cited on page 34.)

[2] Jacob Abernethy, Peter L Bartlett, Alexander Rakhlin, and Ambuj Tewari.
Optimal strategies and minimax lower bounds for online convex games.
2008.
(Cited on page 39.)

[3] Jacob D. Abernethy, Elad Hazan, and Alexander Rakhlin.
Competing in the dark: An efficient algorithm for bandit linear optimization.
In Rocco A. Servedio and Tong Zhang, editors, *21st Annual Conference on Learning Theory - COLT 2008*, pages 263–274. Omnipress, 2008.
(Cited on pages 10 and 25.)

[4] Jacob D Abernethy, Elad Hazan, and Alexander Rakhlin.
Competing in the dark: An efficient algorithm for bandit linear optimization.
2009.
(Cited on pages 30 and 34.)

References II

[5] Baruch Awerbuch and Robert D. Kleinberg.
Adaptive routing with end-to-end feedback: distributed learning and geometric approaches.
In László Babai, editor, *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 45–53. ACM, 2004.
(Cited on page 12.)

[6] Amir Beck and Marc Teboulle.
Mirror descent and nonlinear projected subgradient methods for convex optimization.
Operations Research Letters, 31(3):167–175, 2003.
(Cited on pages 18, 21, and 37.)

[7] David Blackwell.
An analog of the minimax theorem for vector payoffs.
Pacific Journal of Mathematics, 6:1–8, 1956.
(Cited on page 8.)

[8] Nicoló Cesa-Bianchi, Alex Conconi, and Claudio Gentile.
On the generalization ability of on-line learning algorithms.
Advances in neural information processing systems, 14, 2001.
(Cited on page 44.)

References III

[9] Nicolo Cesa-Bianchi and Gábor Lugosi.
Prediction, learning, and games.
Cambridge university press, 2006.
(Cited on pages 41 and 42.)

[10] Thomas M. Cover.
Universal portfolios.
Mathematical Finance, 1(1):1–29, 1991.
(Cited on page 15.)

[11] Yoav Freund and Robert E. Schapire.
A decision-theoretic generalization of on-line learning and an application to boosting.
J. Comput. Syst. Sci., 55(1):119–139, 1997.
(Cited on pages 10, 18, and 19.)

[12] Adam J Grove, Nick Littlestone, and Dale Schuurmans.
General convergence results for linear discriminant updates.
Machine Learning, 43:173–210, 2001.
(Cited on page 30.)

References IV

[13] James Hannan.
4. Approximation to Bayes risk in repeated play, pages 97–140.
Princeton University Press, Princeton.
(Cited on pages 8 and 10.)

[14] Elad Hazan.
Introduction to online convex optimization.
CoRR, abs/1909.05207, 2019.
(Cited on pages 10, 13, and 14.)

[15] A. Kalai and S. Vempala.
Efficient algorithms for universal portfolios.
In *Proceedings 41st Annual Symposium on Foundations of Computer Science*, 2000.
(Cited on page 15.)

[16] Adam Kalai and Santosh Vempala.
Efficient algorithms for online decision problems.
Journal of Computer and System Sciences, 71(3):291–307, 2005.
(Cited on pages 26, 27, and 30.)

References V

[17] Adam Kalai and Santosh S. Vempala.
Efficient algorithms for online decision problems.
In Bernhard Schölkopf and Manfred K. Warmuth, editors, *Computational Learning Theory and Kernel Machines, 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop, COLT/Kernel 2003, Washington, DC, USA, August 24-27, 2003, Proceedings*, volume 2777 of *Lecture Notes in Computer Science*, pages 26–40. Springer, 2003.
(Cited on page 25.)

[18] Jyrki Kivinen and Manfred KK Warmuth.
Relative loss bounds for multidimensional regression problems.
Advances in neural information processing systems, 10, 1997.
(Cited on page 30.)

[19] N. Littlestone and M.K. Warmuth.
The weighted majority algorithm.
In *30th Annual Symposium on Foundations of Computer Science*, 1989.
(Cited on pages 10, 11, and 17.)

[20] Yurii Nesterov.
Primal-dual subgradient methods for convex problems.
Mathematical programming, 120(1):221–259, 2009.
(Cited on pages 18 and 37.)

References VI

[21] Shai Shalev-Shwartz.
Online learning: Theory, algorithms, and applications.
Hebrew University, 2007.
(Cited on pages 30 and 34.)

[22] Shai Shalev-Shwartz et al.
Online learning and online convex optimization.
Foundations and Trends® in Machine Learning, 4(2):107–194, 2012.
(Cited on pages 26 and 27.)

[23] Shai Shalev-Shwartz and Yoram Singer.
A primal-dual perspective of online learning algorithms.
Mach. Learn., 69(2-3):115–142, 2007.
(Cited on page 25.)

[24] Chung-En Tsai, Hao-Chung Cheng, and Yen-Huan Li.
Online self-concordant and relatively smooth minimization, with applications to online portfolio selection and learning quantum states.
CoRR, abs/2210.00997, 2022.
(Cited on page 15.)

References VII

[25] Chung-En Tsai, Hao-Chung Cheng, and Yen-Huan Li.
Online self-concordant and relatively smooth minimization, with applications to online portfolio selection and learning quantum states.
arXiv preprint arXiv:2210.00997, 2022.
(Cited on page 16.)

[26] Volodimir G. Vovk.
Aggregating strategies.
In *Proceedings of the Third Annual Workshop on Computational Learning Theory*, page 371–386, San Francisco, CA, USA, 1990. Morgan Kaufmann Publishers Inc.
(Cited on page 8.)

[27] Martin Zinkevich.
Online convex programming and generalized infinitesimal gradient ascent.
In Tom Fawcett and Nina Mishra, editors, *Machine Learning, Proceedings of the Twentieth International Conference (ICML 2003*, pages 928–936. AAAI Press, 2003.
(Cited on pages 7, 10, 22, and 23.)