Online Learning

— Course Introduction & Syllabus

Joseph Chuang-Chieh Lin

Department of Computer Science & Information Engineering,
Tamkang University

Spring 2023

Credits for the resource

The slides are based on the lectures of Prof. Luca Trevisan:
https://lucatrevisan.github.io/40391/index.html

the lectures of Prof. Shipra Agrawal:
https://ieor8100.github.io/mab/

the lectures of Prof. Francesco Orabona:
https://parameterfree.com/lecture-notes-on-online-learning/
the monograph: https://arxiv.org/abs/1912.13213

and also Elad Hazan's textbook:
Introduction to Online Convex Optimization, 2nd Edition.

- On this course, we will “study together”.
- We rely on the discussions and interactions in the class.
- Sometimes we will use the white board because it’s clearer for illustrating the formulae and ideas step by step.
- We probably follow Prof. Orabona’s textbook.

Topics we plan to cover...

- Introduction & Prerequisites for online learning
- Online (Sub-)Gradient Descent (OGD)
- Online-to-Batch Conversion
- Multiplicative Weight Update (MWU)
- Follow the Regularized Leader (FTRL)
- Online Mirror Descent (OMD)
- Multi-Armed Bandit
- *Extra-Gradient & Optimistic Gradient Descent
- Other selected topics.

Grading Policy

- Attendance (20%)
- Course Interactions (10%)
  - Asking questions (1% for each)
- One Coding Project (10%)
- Midterm Paper/Book Chapter Presentation (30%)
- Final Paper Presentation (30%)

Grading Policy for the Presentations

- Order: According to the seat number in iClass.
- Complete the presentation: 70 point.
  - Duration for each presentation: 30–50 minutes.
- Raising questions: +2 point for each one (maximum +10 point).
- Clearly answering the teacher's 2–4 questions: +5 point for each one.

Grading Policy for the Coding Project

- Work as a team is allowed (3–5 people).
- We will give two options for the project.
  - The easy one: UCB Implementation (5%)
  - The complicated one: Online Portfolio Management Using MWU (or any online algorithms): 10%
- Submit your codes and documentation to iClass.
- One person in each group must present your codes and results in the class.

Outline

- What's online learning?

- What's online learning?
- What about Offline optimization?

Online Convex Optimization

Goal: Design an algorithm such that

- At discrete time steps $t = 1, 2, \ldots$, output $\mathbf{x}_t \in \mathcal{K}$, for each t.
  - $\mathcal{K}$: a convex set of feasible solutions.
- After $\mathbf{x}_t$ is generated, a convex cost function $f_t : \mathcal{K} \mapsto \mathbb{R}$ is revealed.
- Then the algorithm suffers the loss $f_t(\mathbf{x}_t)$.

And we want to minimize the cost.

Online Convex Optimization

Goal: Design an algorithm such that

- At discrete time steps $t = 1, 2, \ldots$, output $\mathbf{x}_t \in \mathcal{K}$, for each t.
  - $\mathcal{K}$: a convex set of feasible solutions.
- After $\mathbf{x}_t$ is generated, a convex cost function $f_t : \mathcal{K} \mapsto \mathbb{R}$ is revealed.
- Then the algorithm suffers the loss $f_t(\mathbf{x}_t)$.

And we want to minimize the cost.

- For example, an adversary chooses $\mathbf{y}_t$ for each t and we suffer the squared difference as the loss $f_t(\mathbf{x}_t) = (\mathbf{x}_t - \mathbf{y}_t)^\top(\mathbf{x}_t - \mathbf{y}_t)$.

The difficulty

- The cost functions f_t could be unknown before t.
- $f_1, f_2, \ldots, f_t, \ldots$ are not necessarily fixed.
  - Can be generated dynamically by an adversary.

What's the regret?

- The offline optimum: After T steps,

$$\min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

- The regret after T steps:

$$\text{regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

What's the regret?

- The offline optimum: After T steps,

$$\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

- The regret after T steps:

$$\text{regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

- The rescue: $\text{regret}_T \leq o(T)$.

What's the regret?

- The offline optimum: After T steps,

$$\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

- The regret after T steps:

$$\text{regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x}).$$

- The rescue: $\text{regret}_T \leq o(T)$. $\Rightarrow$ **No-Regret** in average when $T \to \infty$.
  - For example, $\text{regret}_T / T = \frac{\sqrt{T}}{T} \to 0$ when $T \to \infty$.

Remark

- If an online learning algorithm can guarantee a sublinear regret, it means that its performance, on average, will approach the performance of ANY fixed strategy.
- The regret after T steps with respect to some $\mathbf{u}$:

$$\text{regret}_T(\mathbf{u}) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}).$$

What about comparing dynamic optimum?

- The regret after T steps:

$$\text{dynamic\_regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{z}_1,\mathbf{z}_2,\ldots\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{z}_t).$$

What about comparing dynamic optimum?

- The regret after T steps:

$$\text{dynamic\_regret}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{z}_1, \mathbf{z}_2, \ldots \in \mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{z}_t).$$

- What's the difficulty & the issue?

The best in the hindsight vs. follow the leader (1/4)

- Let $\mathbf{x}_T^* := \arg\min \sum_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x})$

The best in the hindsight vs. follow the leader (1/4)

- Let $\mathbf{x}_T^* := \arg\min \sum_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) = \arg\min \sum_{\mathbf{x}\in\mathcal{K}} (\mathbf{x}-\mathbf{y}_t)^\top(\mathbf{x}-\mathbf{y}_t)$.
  - The hindsight optimum.

The best in the hindsight vs. follow the leader (1/4)

- Let $\mathbf{x}_T^* := \arg\min \sum_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) = \arg\min \sum_{\mathbf{x}\in\mathcal{K}} (\mathbf{x} - \mathbf{y}_t)^\top (\mathbf{x} - \mathbf{y}_t)$.
  - The hindsight optimum.
- Let's say that we guess on each round t by

$$\mathbf{x}_t = \mathbf{x}_{t-1}^* = \frac{1}{t-1} \sum_{t=1}^{t-1} \mathbf{y}_t.$$

The best in the hindsight vs. follow the leader (2/4)

Lemma

Let $V \subseteq \mathbb{R}^d$ and let $\ell_t : V \mapsto \mathbb{R}$ be an arbitrary sequence of loss functions. Denote by $\mathbf{x}_t^*$ a minimizer of the cumulative losses over the previous t rounds in V. Then, we have

$$\sum_{t=1}^{T} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T} \ell_t(\mathbf{x}_T^*).$$

The best in the hindsight vs. follow the leader (2/4)

Lemma

Let $V \subseteq \mathbb{R}^d$ and let $\ell_t : V \mapsto \mathbb{R}$ be an arbitrary sequence of loss functions. Denote by $\mathbf{x}_t^*$ a minimizer of the cumulative losses over the previous t rounds in V. Then, we have

$$\sum_{t=1}^{T} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T} \ell_t(\mathbf{x}_T^*).$$

- We prove the theorem by induction on T.

The best in the hindsight vs. follow the leader (2/4)

Lemma

Let $V \subseteq \mathbb{R}^d$ and let $\ell_t : V \mapsto \mathbb{R}$ be an arbitrary sequence of loss functions. Denote by $\mathbf{x}_t^*$ a minimizer of the cumulative losses over the previous t rounds in V. Then, we have

$$\sum_{t=1}^{T} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T} \ell_t(\mathbf{x}_T^*).$$

- We prove the theorem by induction on T.
- The base case ($T = 1$) is true. (WHY?)

The best in the hindsight vs. follow the leader (2/4)

Lemma

Let $V \subseteq \mathbb{R}^d$ and let $\ell_t : V \mapsto \mathbb{R}$ be an arbitrary sequence of loss functions. Denote by $\mathbf{x}_t^*$ a minimizer of the cumulative losses over the previous t rounds in V. Then, we have

$$\sum_{t=1}^{T} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T} \ell_t(\mathbf{x}_T^*).$$

- We prove the theorem by induction on T.
- The base case ($T = 1$) is true. (WHY?)

$$\ell_1(\mathbf{x}_1^*) \leq \ell_1(\mathbf{x}_1)$$

The best in the hindsight vs. follow the leader (3/4)

- For $T \geq 2$, we assume that $\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_{T-1}^*)$.
  - Induction hypothesis.
- Note that

$$\sum_{t=1}^{T} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T} \ell_t(\mathbf{x}_T^*)$$

is equivalent to

$$\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).$$

(WHY?)

The best in the hindsight vs. follow the leader (4/4)

- So to prove

$$\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).$$

The best in the hindsight vs. follow the leader (4/4)

- So to prove

$$\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \le \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).$$

by induction hypothesis we have

$$\begin{aligned} \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) &\le \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_{T-1}^*) \\ &\le \end{aligned}$$

The best in the hindsight vs. follow the leader (4/4)

- So to prove
$$\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).$$

 by induction hypothesis we have
$$\begin{aligned}\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) &\leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_{T-1}^*) \\ &\leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).\end{aligned}$$

The best in the hindsight vs. follow the leader (4/4)

- So to prove
$$\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) \leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).$$
by induction hypothesis we have
$$\begin{aligned}\sum_{t=1}^{T-1} \ell_t(\mathbf{x}_t^*) &\leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_{T-1}^*) \\ &\leq \sum_{t=1}^{T-1} \ell_t(\mathbf{x}_T^*).\end{aligned}$$
- The lemma is proved.

An Example of Sublinear-Regret (1/4)

Consider one-dimensional $x_t, y_t \in \mathbb{R}$ to simplify our discussion.

Theorem

Let $y_t \in [0,1]$ for $t = 1, 2, \ldots, T$ be an arbitrary sequence of numbers. Suppose that the algorithm outputs $x_t = x^*_{t-1} = \frac{1}{t-1}\sum_{i=1}^{t-1} y_i$. Then, we have

$$\sum_{t=1}^{T}(x_t - y_t)^2 - \min_{x\in[0,1]}\sum_{t=1}^{T}(x - y_t)^2 \leq 4 + 4\ln T.$$

- Use previous lemma to "upper bound the regret".

An Example of Sublinear-Regret (2/4)

$$
\begin{aligned}
\sum_{t=1}^{T}(x_t - y_t)^2 - \min_{x\in[0,1]}\sum_{t=1}^{T}(x - y_t)^2 &= \sum_{t=1}^{T}(x^*_{t-1} - y_t)^2 - \sum_{t=1}^{T}(x^*_T - y_t)^2 \\
&\le \sum_{t=1}^{T}(x^*_{t-1} - y_t)^2 - \sum_{t=1}^{T}(x^*_t - y_t)^2.
\end{aligned}
$$

An Example of Sublinear-Regret (3/4)

Note that

$$
\begin{aligned}
(x^*_{t-1} - y_t)^2 - (x^*_t - y_t)^2 &= (x^*_{t-1})^2 - 2y_t x^*_{t-1} - (x^*_t)^2 + 2y_t x^*_t \\
&= (x^*_{t-1} + x^*_t - 2y_t) \cdot (x^*_{t-1} - x^*_t)
\end{aligned}
$$

An Example of Sublinear-Regret (3/4)

Note that

$$
\begin{aligned}
(x_{t-1}^* - y_t)^2 - (x_t^* - y_t)^2 &= (x_{t-1}^*)^2 - 2y_t x_{t-1}^* - (x_t^*)^2 + 2y_t x_t^* \\
&= (x_{t-1}^* + x_t^* - 2y_t) \cdot (x_{t-1}^* - x_t^*) \\
&\leq |x_{t-1}^* + x_t^* - 2y_t| \cdot |x_{t-1}^* - x_t^*|
\end{aligned}
$$

An Example of Sublinear-Regret (3/4)

Note that

$$
\begin{aligned}
(x_{t-1}^* - y_t)^2 - (x_t^* - y_t)^2 &= (x_{t-1}^*)^2 - 2y_t x_{t-1}^* - (x_t^*)^2 + 2y_t x_t^* \\
&= (x_{t-1}^* + x_t^* - 2y_t) \cdot (x_{t-1}^* - x_t^*) \\
&\leq |x_{t-1}^* + x_t^* - 2y_t| \cdot |x_{t-1}^* - x_t^*| \\
&\leq 2|x_{t-1}^* - x_t^*| \\
&= 2\left| \frac{1}{t-1} \sum_{i=1}^{t-1} y_i - \frac{1}{t} \sum_{i=1}^{t} y_i \right|
\end{aligned}
$$

An Example of Sublinear-Regret (3/4)

Note that

$$
\begin{aligned}
(x^*_{t-1} - y_t)^2 - (x^*_t - y_t)^2 &= (x^*_{t-1})^2 - 2y_t x^*_{t-1} - (x^*_t)^2 + 2y_t x^*_t \\
&= (x^*_{t-1} + x^*_t - 2y_t) \cdot (x^*_{t-1} - x^*_t) \\
&\leq |x^*_{t-1} + x^*_t - 2y_t| \cdot |x^*_{t-1} - x^*_t| \\
&\leq 2|x^*_{t-1} - x^*_t| \\
&= 2\left| \frac{1}{t-1} \sum_{i=1}^{t-1} y_i - \frac{1}{t} \sum_{i=1}^{t} y_i \right| \\
&= 2\left| \left(\frac{1}{t-1} - \frac{1}{t} \right) \sum_{i=1}^{t-1} y_i - \frac{y_t}{t} \right| \\
&\leq 2\left| \frac{1}{t(t-1)} \sum_{i=1}^{t-1} y_i \right| + \frac{2|y_t|}{t} \leq \frac{2}{t} + \frac{2|y_t|}{t} \leq \frac{4}{t}.
\end{aligned}
$$

An Example of Sublinear-Regret (4/4)

Overall, we have

$$\begin{aligned}
\sum_{t=1}^{T}(x_t - y_t)^2 - \min_{x\in[0,1]} \sum_{t=1}^{T}(x - y_t)^2 &\leq 4\sum_{t=1}^{T}\frac{1}{t} \\
&\leq 1 + \int_2^{T+1} \frac{1}{t-1}\,dt \\
&= 1 + \ln T. \\
&\quad (\text{ or simply } O(\ln T)).
\end{aligned}$$

An Example of Sublinear-Regret (4/4)

Overall, we have

$$
\begin{aligned}
\sum_{t=1}^{T}(x_t - y_t)^2 - \min_{x\in[0,1]} \sum_{t=1}^{T}(x - y_t)^2 &\leq 4\sum_{t=1}^{T}\frac{1}{t} \\
&\leq 1 + \int_{2}^{T+1} \frac{1}{t-1}\,dt \\
&= 1 + \ln T. \\
&\quad (\text{ or simply } O(\ln T)).
\end{aligned}
$$

- No parameters are required to tune (e.g., learning rates, regularization terms, etc.).
- It doesn't make sense either to have such parameters because we cannot run the algorithm over the data multiple times!

Exercise 01

- Show that $\sum_{t=1}^{T} \frac{1}{\sqrt{t}} \leq 2\sqrt{T} - 1$.

Exercise 02

- Extend the algorithm and the analysis to the case when adversary selects a vector $\mathbf{y}_t \in \mathbb{R}^d$ such that
  - $\|\mathbf{y}_t\|_2 \leq 1$,
  - the algorithm selects $\mathbf{x}_t \in \mathbb{R}^d$, and
  - the loss function is $\|\mathbf{x}_t - \mathbf{y}_t\|_2^2$.
- Prove an upper bound to the regret $O(\log T)$ which does not depend on d.

Hint: Using Cauchy-Schwarz inequality: $\langle \mathbf{x}, \mathbf{y} \rangle \leq \|\mathbf{x}\|_2 \|\mathbf{y}\|_2$.

Online learning applications

- Click prediction.
- Portfolio weight adjustment.
- Routing on a network.
- Convergence to an equilibrium for iterative/repeated games.

Regret & profitability

- We try to optimize the regret.

Regret & profitability

- We try to optimize the regret.
- Yet, like the scenario of online portfolio adjustment, does the regret corresponds to definite PnL?

Prerequisites (1/7)

Diameter

Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a bounded convex and closed set in Euclidean space. We denote by D an upper bound on the diameter of $\mathcal{K}$:

$$\forall \mathbf{x}, \mathbf{y} \in \mathcal{K}, \|\mathbf{x} - \mathbf{y}\| \leq D.$$

Convex set

A set $\mathcal{K}$ is convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$, we have

$$\forall \alpha \in [0, 1], \alpha \mathbf{x} + (1 - \alpha)\mathbf{y} \in \mathcal{K}.$$

Prerequisites (2/7)

Convex function

A function $f : \mathcal{K} \mapsto \mathbb{R}$ is convex if for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$,

$$\forall \alpha \in [0, 1], f((1 - \alpha)\mathbf{x} + \alpha \mathbf{y}) \leq (1 - \alpha) f(\mathbf{x}) + \alpha f(\mathbf{y}).$$

Equivalently, if f is differentiable (i.e., $\nabla f(\mathbf{x})$ exists for all $\mathbf{x} \in \mathcal{K}$), then f is convex if and only if for all $\mathbf{x}, \mathbf{y} \in \mathcal{K}$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}).$$

Prerequisites (3/7)

Theorem [Rockafellar 1970]

Suppose that $f : \mathcal{K} \mapsto \mathbb{R}$ is a convex function and let $\mathbf{x} \in \text{int dom}(f)$. If f is differentiable at $\mathbf{x}$, then for all $\mathbf{y} \in \mathbb{R}^d$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle.$$

Subgradient

For a function $f : \mathbb{R}^d \mapsto \mathbb{R}$, $\mathbf{g} \in \mathbb{R}^d$ is a subgradient of f at $x \in \mathbb{R}^d$ if for all $\mathbf{y} \in \mathbb{R}^d$,

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \langle \mathbf{g}, \mathbf{y} - \mathbf{x} \rangle.$$

Prerequisites (4/7)

Projection

The closest point of $\mathbf{y}$ in a convex set $\mathcal{K}$ in terms of norm $\|\cdot\|$:

$$\Pi_{\mathcal{K}}(\mathbf{y}) := \arg\min_{\mathbf{x}\in\mathcal{K}} \|\mathbf{x}-\mathbf{y}\|.$$

Pythagoras Theorem

Let $\mathcal{K} \subseteq \mathbb{R}^d$ be a convex set, $\mathbf{y} \in \mathbb{R}^d$ and $\mathbf{x} = \Pi_{\mathcal{K}}(\mathbf{y})$. Then for any $\mathbf{z} \in \mathcal{K}$, we have

$$\|\mathbf{y}-\mathbf{z}\| \geq \|\mathbf{x}-\mathbf{z}\|.$$

Prerequisites (5/7)

Minimum vs. zero gradient

$$\nabla f(\mathbf{x}) = 0 \text{ iff } \mathbf{x} \in \arg\min_{\mathbf{x}\in\mathbb{R}^d} \{f(\mathbf{x})\}.$$

First-Order Optimality Condition for Convex Functions

Let

- $\mathcal{K} \subseteq \mathbb{R}^d$ be a convex set,
- f be a convex function which is differentiable over an open set that contains $\mathcal{K}$, and
- $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{K}} f(\mathbf{x})$,

then for any $\mathbf{y} \in \mathcal{K}$ we have

$$\nabla f(\mathbf{x}^*)^\top (\mathbf{y} - \mathbf{x}^*) \geq 0.$$

Prerequisites (6/7)

Jensen's Inequality

Let $f : \mathbb{R}^d \mapsto (-\infty, +\infty]$ be a measurable convex function and $\mathbf{x}$ be an $\mathbb{R}^d$-valued random variable such that $\mathbf{E}[\mathbf{x}]$ exists and $\mathbf{x} \in \mathrm{dom}(f)$ with probability 1. Then,

$$\mathbf{E}[f(\mathbf{x})] \geq f(\mathbf{E}[\mathbf{x}]).$$

Prerequisites (7/7)

Cauchy-Schwarz inequality

For all vectors $\mathbf{u}$ and $\mathbf{v}$ of an inner product space,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle.$$

or equivalently,

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \cdot \|\mathbf{v}\|.$$

Prerequisites (7/7)

Cauchy-Schwarz inequality

For all vectors $\mathbf{u}$ and $\mathbf{v}$ of an inner product space,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle.$$

or equivalently,

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \cdot \|\mathbf{v}\|.$$

- Let's have a look at an research example.

Prerequisites (7/7)

Cauchy-Schwarz inequality

For all vectors $\mathbf{u}$ and $\mathbf{v}$ of an inner product space,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle.$$

or equivalently,

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \cdot \|\mathbf{v}\|.$$

- Let's have a look at an research example.

$$SW(\mathbf{s}) = \sum_{i \in [m]} \frac{u(s_i)}{\sum_{j \in [m]} u(s_j)} \cdot u(s_i) = \frac{\sum_{i \in [m]} u(s_i)^2}{\sum_{j \in [m]} u(s_j)}$$

Prerequisites (7/7)

Cauchy-Schwarz inequality

For all vectors **u** and **v** of an inner product space,

$$|\langle \mathbf{u}, \mathbf{v} \rangle|^2 \leq \langle \mathbf{u}, \mathbf{u} \rangle \cdot \langle \mathbf{v}, \mathbf{v} \rangle.$$

or equivalently,

$$|\langle \mathbf{u}, \mathbf{v} \rangle| \leq \|\mathbf{u}\| \cdot \|\mathbf{v}\|.$$

- Let's have a look at an research example.

$$\begin{aligned} SW(\mathbf{s}) &= \sum_{i \in [m]} \frac{u(s_i)}{\sum_{j \in [m]} u(s_j)} \cdot u(s_i) = \frac{\sum_{i \in [m]} u(s_i)^2}{\sum_{j \in [m]} u(s_j)} \\ &\geq \frac{1}{m} \cdot \sum_{i \in [m]} u(s_i). \end{aligned}$$

Convex losses to linear losses

- We have the convex loss function $f_t(\mathbf{x}_t)$ at time t.
- Say we have subgradients $\mathbf{g}_t$ for each $\mathbf{x}_t$.
- $f(\mathbf{x}_t) - f(\mathbf{u}) \leq \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle$ for each $\mathbf{u} \in \mathbb{R}^d$.

Convex losses to linear losses

- We have the convex loss function $f_t(\mathbf{x}_t)$ at time t.
- Say we have subgradients $\mathbf{g}_t$ for each $\mathbf{x}_t$.
- $f(\mathbf{x}_t) - f(\mathbf{u}) \leq \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle$ for each $\mathbf{u} \in \mathbb{R}^d$.
- Hence, if we define $\tilde{f}_t(\mathbf{x}) := \langle \mathbf{g}_t, \mathbf{x} \rangle$, then for any $\mathbf{u} \in \mathbb{R}^d$,

$$\sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f(\mathbf{u})) \leq \sum_{t=1}^{T} \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle = \sum_{t=1}^{T} \tilde{f}_t(\mathbf{x}_t) - \tilde{f}(\mathbf{u}).$$

Convex losses to linear losses

- We have the convex loss function $f_t(\mathbf{x}_t)$ at time t.
- Say we have subgradients $\mathbf{g}_t$ for each $\mathbf{x}_t$.
- $f(\mathbf{x}_t) - f(\mathbf{u}) \le \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle$ for each $\mathbf{u} \in \mathbb{R}^d$.
- Hence, if we define $\tilde{f}_t(\mathbf{x}) := \langle \mathbf{g}_t, \mathbf{x} \rangle$, then for any $\mathbf{u} \in \mathbb{R}^d$,

$$\sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f(\mathbf{u})) \le \sum_{t=1}^{T} \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle = \sum_{t=1}^{T} \tilde{f}_t(\mathbf{x}_t) - \tilde{f}(\mathbf{u}).$$

  - Note that $\langle \mathbf{u}, \mathbf{v} + \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{v} \rangle + \langle \mathbf{u}, \mathbf{w} \rangle$.

Convex losses to linear losses

- We have the convex loss function $f_t(\mathbf{x}_t)$ at time t.
- Say we have subgradients $\mathbf{g}_t$ for each $\mathbf{x}_t$.
- $f(\mathbf{x}_t) - f(\mathbf{u}) \leq \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle$ for each $\mathbf{u} \in \mathbb{R}^d$.
- Hence, if we define $\tilde{f}_t(\mathbf{x}) := \langle \mathbf{g}_t, \mathbf{x} \rangle$, then for any $\mathbf{u} \in \mathbb{R}^d$,

$$\sum_{t=1}^{T} (f_t(\mathbf{x}_t) - f(\mathbf{u})) \leq \sum_{t=1}^{T} \langle \mathbf{g}, \mathbf{x_t} - \mathbf{u} \rangle = \sum_{t=1}^{T} \tilde{f}_t(\mathbf{x}_t) - \tilde{f}(\mathbf{u}).$$

  - Note that $\langle \mathbf{u}, \mathbf{v} + \mathbf{w} \rangle = \langle \mathbf{u}, \mathbf{v} \rangle + \langle \mathbf{u}, \mathbf{w} \rangle$.

⋆ OCO → OLO

Remark

- The reduction implies that we can build online (convex optimization) algorithms that deal only with linear losses.
- Note that this reduction isn't always optimal.
- Yet, it allows us to easily construct OCO algorithms in many cases.

Discussions