Universal Online Learning with Gradient-Variation Regret

Peng Zhao

School of AI, Nanjing University

Dec. 18th @ UCSB

Outline

- Background
- Motivation
- Our Approach
- Conclusion

Outline

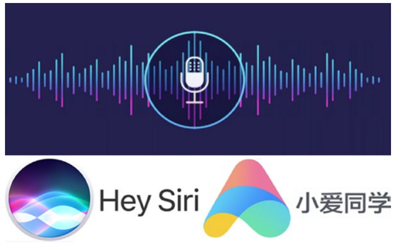
- Background
- Motivation
- Our Approach
- Conclusion

Machine Learning

- Machine Learning has achieved great success in recent years.

image recognition

search engine



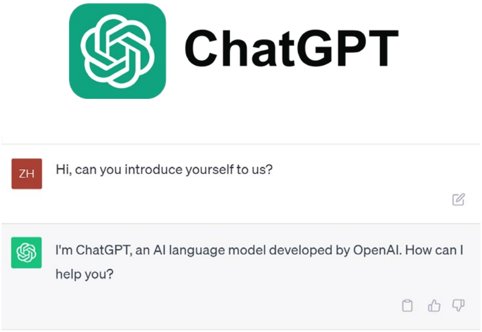
voice assistant

recommendation

AlphaGo Games

automatic driving

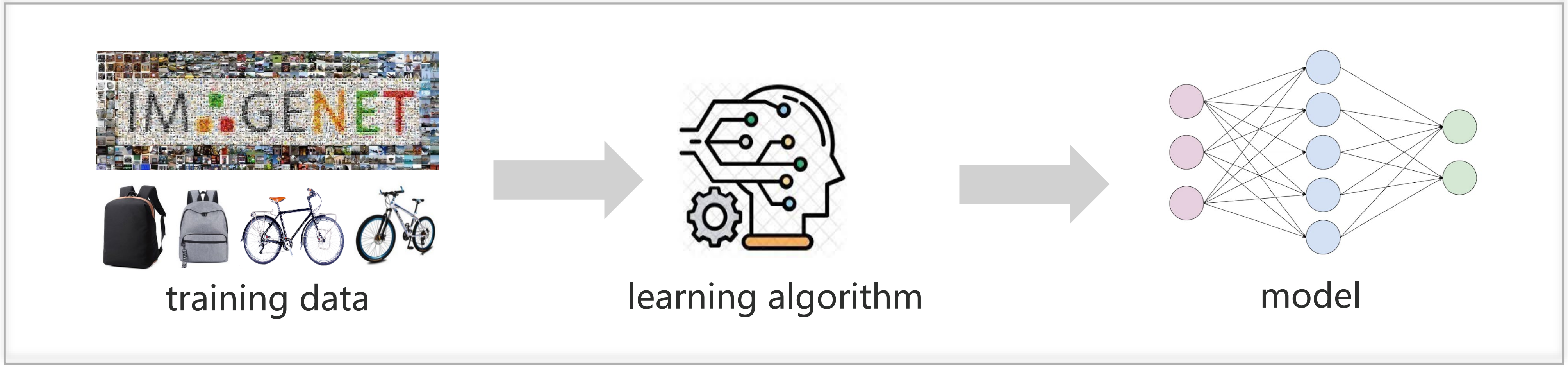
medical diagnosis



large language model

Machine Learning

- A standard pipeline for machine learning deployments.



- Learning as optimization: using ERM to learn the model

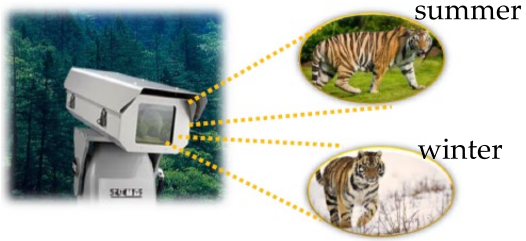
$$\min_{\mathbf{x}\in\mathcal{X}} \sum_{i=1}^{m} \ell(\mathbf{x}; z_i)$$

learning the model based on the (offline) training dataset $S = \{z_1, \ldots, z_m\}$

Online Learning

- In many applications, data are coming in an ***online*** fashion

species monitoring



urban computing

route planning

- Online learning/optimization

  - update the model in an iterated optimization fashion

  - need to have guarantees for the online update

Online Convex Optimization (OCO)

- View online learning as a game between *learner* and *environment*.

At each round $t = 1, 2, \ldots, T$:

- the learner submits $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$
- at the same time, environments decide a convex loss function f_t
- the learner suffers $f_t(\mathbf{x}_t)$ and receives gradient information

- **Regret**: online prediction as good as the best offline model

$$\mathrm{Reg}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

The learner's excess loss compared to the best offline model in hindsight.

Online Convex Optimization (OCO)

- **Regret**: online prediction as good as the best offline model

$$\mathrm{Reg}_T = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

- There are plenty of prior efforts for regret minimization.
  - Online Gradient Descent (OGD)

$$\mathbf{x}_{t+1} = \Pi_{\mathcal{X}}\left[\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t)\right]$$

where $\Pi_{\mathcal{X}}[\cdot]$ denotes the Euclidean projection onto feasible domain $\mathcal{X}$.

  - Other frameworks include online mirror descent and FTRL.

Outline

- Background
- Motivation
- Our Approach
- Conclusion

OCO: classic methods

- **Classic Methods:** require knowing the ***function curvature*** and obtain ***worst-case*** regret guarantees

Function type	Algorithm	Regret
convex	Online Gradient Descent with $\eta_t \approx \frac{1}{\sqrt{t}}$	$\mathcal{O}(\sqrt{T})$
λ-strongly convex	Online Gradient Descent with $\eta_t = \frac{1}{\lambda t}$	$\mathcal{O}(\log T)$
α-exp-concave	Online Newton Step with α	$\mathcal{O}(d \log T)$

Recent studies explore ***two levels of adaptivity***.

- **High-Level:** adaptive to ***unknown function curvatures***
- **Low-Level:** adaptive to ***unknown niceness of environments***

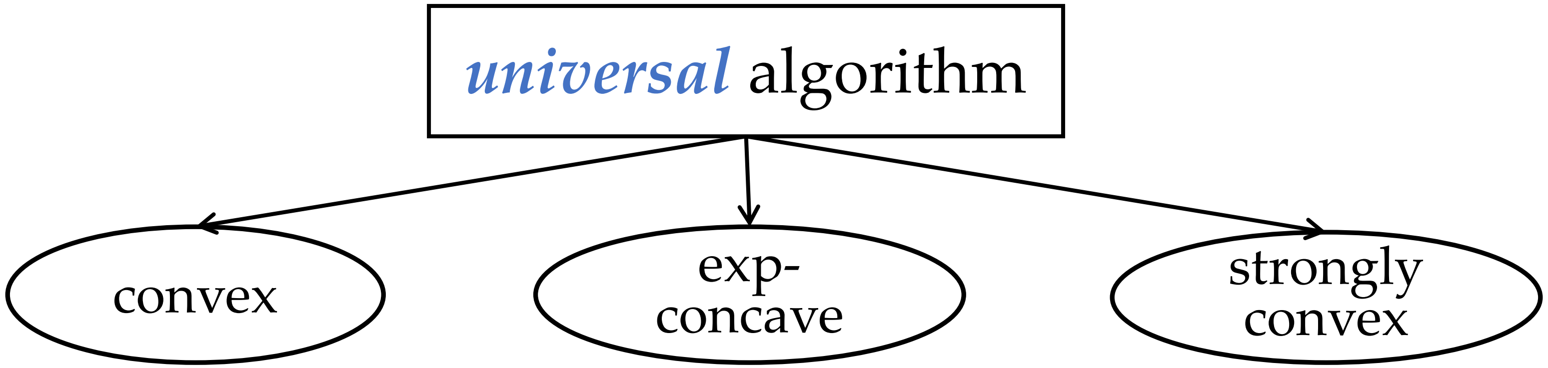
OCO: high-level adaptivity

- **High-Level:** adaptive to ***unknown function curvatures***

  Universal method aims to develop a single algorithm for different families:

  (i) agnostic to the specific function curvature;

  (ii) while achieving the same regret as if they were known.



⇨ An algorithm achieves $\mathcal{O}(\sqrt{T})$, $\mathcal{O}(d \log T)$, and $\mathcal{O}(\log T)$ regret for convex/exp-concave/str. convex functions, respectively.

OCO: low-level adaptivity

- **Low-Level:** adaptive to ***unknown niceness of environments***

Problem-dependent method aims to develop more adaptive bounds:

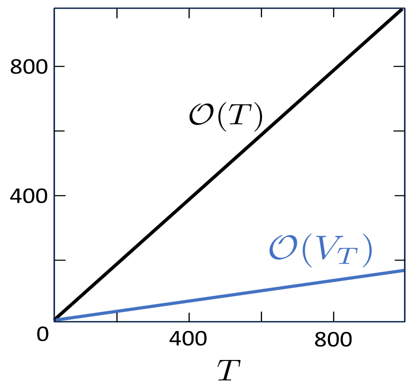
(i) regret guarantee can be substantially improved for easy environments ;

(ii) while can simultaneously safeguard the worst-case minimax rate.

Gradient variation:

$$V_T \triangleq \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$$

measure the cumulative variations in gradients



⇨ Improved regret of $\mathcal{O}(\sqrt{V_T})$, $\mathcal{O}(d\log V_T)$, and $\mathcal{O}(\log V_T)$ can be attained for convex/exp-concave/str. convex functions, respectively (using different algorithms).

Guiding Question

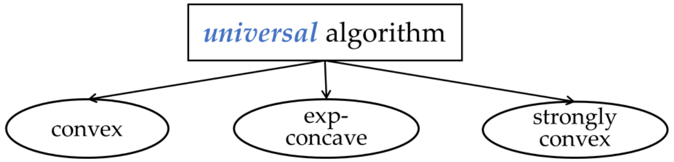
Is it possible to design an algorithm with two-level adaptivity?

i.e., universal to function curvature, and adaptive to gradient variations

OCO: high-level adaptivity

- **High-Level:** adaptive to *unknown function curvatures*

Universal method aims to develop a single algorithm for different families:
(i) agnostic to the specific function curvature;
(ii) while achieving the same regret as if they were known.



⇨ An algorithm achieves $\mathcal{O}(\sqrt{T})$, $\mathcal{O}(d \log T)$, and $\mathcal{O}(\log T)$ regret for convex/exp-concave/str. convex functions, respectively.

OCO: low-level adaptivity

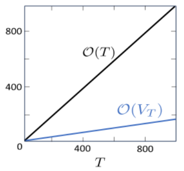
- **Low-Level:** adaptive to *unknown niceness of environments*

Problem-dependent method aims to develop more adaptive bounds:
(i) regret guarantee can be substantially improved for easy environments;
(ii) while can simultaneously safeguard the worst-case minimax rate.

Gradient variation:

$$V_T \triangleq \sum_{t=2}^{T} \sup_{\mathbf{x} \in \mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$$

measure the cumulative variations in gradients



⇨ Improved regret of $\mathcal{O}(\sqrt{V_T})$, $\mathcal{O}(d \log V_T)$, and $\mathcal{O}(\log V_T)$ can be attained for convex/exp-concave/str. convex functions, respectively (using different algorithms).

Main Result

- We provide an affirmative answer by providing the following result.

Theorem 1 (Yan-Z-Zhou; NeurIPS 2023). *Under standard assumptions, our algorithm ensures that*

- *it achieves* $\mathcal{O}(\log V_T)$ *regret for strongly convex functions;*
- *it achieves* $\mathcal{O}(d \log V_T)$ *regret for exp-concave functions;*
- *it achieves* $\widehat{\mathcal{O}}(\sqrt{V_T})$ *regret for convex functions.*

Here, $V_T = \sum_{t=2}^{T} \sup_{\mathbf{x} \in \mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$ *is gradient variation and* $\widehat{\mathcal{O}}(\cdot)$ *omits* $\log V_T$ *factors.*

A single algorithm with simultaneously near-optimal gradient-variation regret bounds for convex/exp-concave/strongly convex functions.

Why Gradient Variation?

- **Importance in Theory and Practice:**
  - *Exploiting the niceness of environments, while safeguarding the minimax rate*
    - V_T denotes the variation in gradients that can be much smaller than $\mathcal{O}(T)$.
    - Gradient-variation regret bounds $\mathcal{O}(\log V_T)$, $\mathcal{O}(d \log V_T)$, and $\mathcal{O}(\sqrt{V_T})$ can recover the minimax rate of $\mathcal{O}(\log T)$, $\mathcal{O}(d \log T)$, and $\mathcal{O}(\sqrt{T})$.
  - *Implications in **Games** & **Stochastic Optimization***
    - Gradient variation bounds are essential for obtaining fast rates in games.
    - Gradient variation can bridge stochastic and adversarial optimization.

Implications: Games

- **Gradient Variation in *Games***: [Syrgkanis et al., NIPS'15]

Example: 

y-player decision $\mathbf{y}_t = ({}^1/_2 \ {}^1/_2 \ \ 0)^\top$

x-player decision $\mathbf{x}_t = \begin{pmatrix} 0 \\ {}^1/_2 \\ {}^1/_2 \end{pmatrix}$

Game matrix A

	Rock	Scissors	Paper
Rock	$(0,0)$	$(1,-1)$	$(-1,1)$
Scissors	$(-1,1)$	$(0,0)$	$(-1,1)$
Paper	$(1,-1)$	$(-1,1)$	$(0,0)$

Implications: Games

- **Gradient Variation in *Games*:** [Syrgkanis et al., NIPS'15]

Online Game Protocol

The environments decide a payoff matrix A

At each round $t = 1, 2, \ldots, T$:

- x-player submits $\mathbf{x}_t \in \Delta_d$ and y-player submits $\mathbf{y}_t \in \Delta_d$
- the x-player suffers loss $\mathbf{x}_t^\top A \mathbf{y}_t$ and receives gradient $A\mathbf{y}_t$, the y-player receives reward $\mathbf{x}_t^\top A \mathbf{y}_t$ and receives gradient $A\mathbf{x}_t$

Gradient-variation online learning plays an important role in games.

Implications: Games

Deploying *gradient-variation algorithm* (e.g., online mirror descent with last-round gradient) attains:

$$f_t^x(\mathbf{x}) \triangleq \mathbf{x}^\top A \mathbf{y}_t$$

$$f_{t-1}^x(\mathbf{x}) \triangleq \mathbf{x}^\top A \mathbf{y}_{t-1}$$

$$\mathrm{Reg}_T^x \lesssim 1 + \underbrace{\sum_{t=2}^T \|A\mathbf{y}_t - A\mathbf{y}_{t-1}\|_\infty^2}_{\text{gradient variation}} \underbrace{- \sum_{t=2}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_1^2}_{\text{negative stability}}$$

Deploying *gradient-variation algorithm* (e.g., online mirror descent with last-round gradient) attains:

$$f_t^y(\mathbf{y}) \triangleq \mathbf{x}_t^\top A \mathbf{y}$$

$$f_{t-1}^y(\mathbf{y}) \triangleq \mathbf{x}_{t-1}^\top A \mathbf{y}$$

$$\mathrm{Reg}_T^y \lesssim 1 + \underbrace{\sum_{t=2}^T \|\mathbf{x}_t^\top A - \mathbf{x}_{t-1}^\top A\|_\infty^2}_{\text{gradient variation}} \underbrace{- \sum_{t=2}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|_1^2}_{\text{negative stability}}$$

***Regret summation** is usually related to some global performance measures in games, such as Nash equilibrium regret and duality gap.*

Implications: Games

Deploying *gradient-variation algorithm* (e.g., online mirror descent with last-round gradient) attains:

$$f_t^x(\mathbf{x}) \triangleq \mathbf{x}^\top A \mathbf{y}_t$$

$$f_{t-1}^x(\mathbf{x}) \triangleq \mathbf{x}^\top A \mathbf{y}_{t-1}$$

$$\text{Reg}_T^x \lesssim 1 + \underbrace{\sum_{t=2}^T \|A\mathbf{y}_t - A\mathbf{y}_{t-1}\|_\infty^2}_{\text{gradient variation}} \underbrace{- \sum_{t=2}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_1^2}_{\text{negative stability}}$$

Deploying *gradient-variation algorithm* (e.g., online mirror descent with last-round gradient) attains:

$$f_t^y(\mathbf{y}) \triangleq \mathbf{x}_t^\top A \mathbf{y}$$

$$f_{t-1}^y(\mathbf{y}) \triangleq \mathbf{x}_{t-1}^\top A \mathbf{y}$$

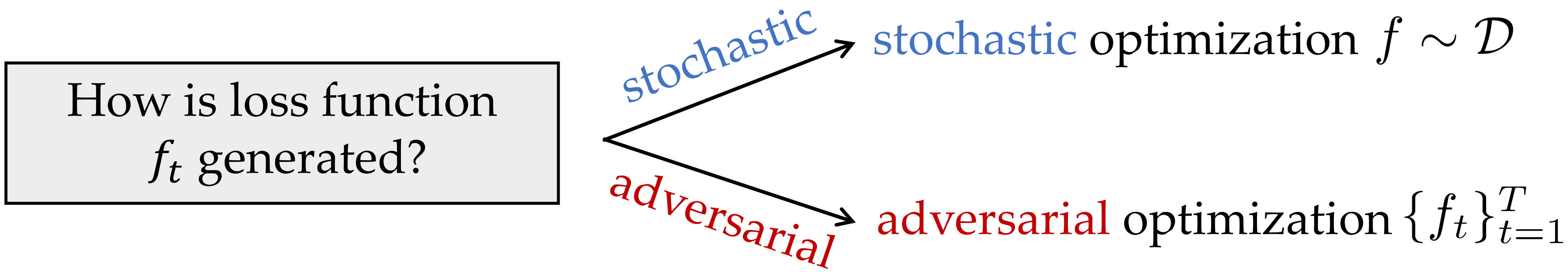
$$\text{Reg}_T^y \lesssim 1 + \underbrace{\sum_{t=2}^T \|\mathbf{x}_t^\top A - \mathbf{x}_{t-1}^\top A\|_\infty^2}_{\text{gradient variation}} \underbrace{- \sum_{t=2}^T \|\mathbf{y}_t - \mathbf{y}_{t-1}\|_1^2}_{\text{negative stability}}$$

$\Rightarrow \text{Reg}_T^x + \text{Reg}_T^y \leq \mathcal{O}(1)$ *which is essential for the* $\mathcal{O}\left(\frac{1}{T}\right)$ *fast rate in games.*

Implications: Stochastic Opt.

- **Gradient Variation in *Stochastic/Adversarial Optimization* :**

[Sachs et al., NeurIPS'22]



- The studies on these two fields are previously ***separate***.
- Recent works reveal the essential role of ***gradient variation***, which provides an important interpolation between stochastic and adversarial optimization.

Implications: Stochastic Opt.

- **SEA (Stochastically Extended Adversarial) model** [Sachs et al., NeurIPS'22]

  Setup: at round $t \in [T]$, SEA optimizes $\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$

  f_t is the *randomized function* sampled from underlying distribution $\mathcal{D}_t$: $f_t \sim \mathcal{D}_t$

  F_t is the *expected function* of f_t: $F_t(\cdot) \triangleq \mathbb{E}_{f_t\sim\mathcal{D}_t}[f_t(\cdot)]$

Two crucial complexity measures:

$$\sigma_{1:T}^2 \triangleq \sum_{t=1}^{T} \max_{\mathbf{x}\in\mathcal{X}} \mathbb{E}_{f_t\sim\mathcal{D}_t}[\|\nabla f_t(\mathbf{x}) - \nabla F_t(\mathbf{x})\|^2], \quad \Sigma_{1:T}^2 \triangleq \mathbb{E}\left[\sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla F_t(\mathbf{x}) - \nabla F_{t-1}(\mathbf{x})\|^2\right]$$

stochastic change — *adversarial change*

Implications: Stochastic Opt.

- **SEA (Stochastically Extended Adversarial) model** [Sachs et al., NeurIPS'22]

Setup: at round $t \in [T]$, SEA optimizes $\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$

f_t is the *randomized function* sampled from underlying distribution $\mathcal{D}_t$: $f_t \sim \mathcal{D}_t$

F_t is the *expected function* of f_t: $F_t(\cdot) \triangleq \mathbb{E}_{f_t\sim\mathcal{D}_t}[f_t(\cdot)]$

⇨ SEA model can be solved by deploying gradient-variation algorithm over the randomized function $\{f_t\}_{t=1}^T$.

$$\underbrace{\nabla f_t(\mathbf{x})-\nabla f_{t-1}(\mathbf{x})}_{\text{gradient variation}} = \underbrace{[\nabla f_t(\mathbf{x})-\nabla F_t(\mathbf{x})]}_{\text{stochastic change}}+\underbrace{[\nabla F_t(\mathbf{x})-\nabla F_{t-1}(\mathbf{x})]}_{\text{adversarial change}}+\underbrace{[\nabla F_{t-1}(\mathbf{x})-\nabla f_{t-1}(\mathbf{x})]}_{\text{stochastic change}}$$

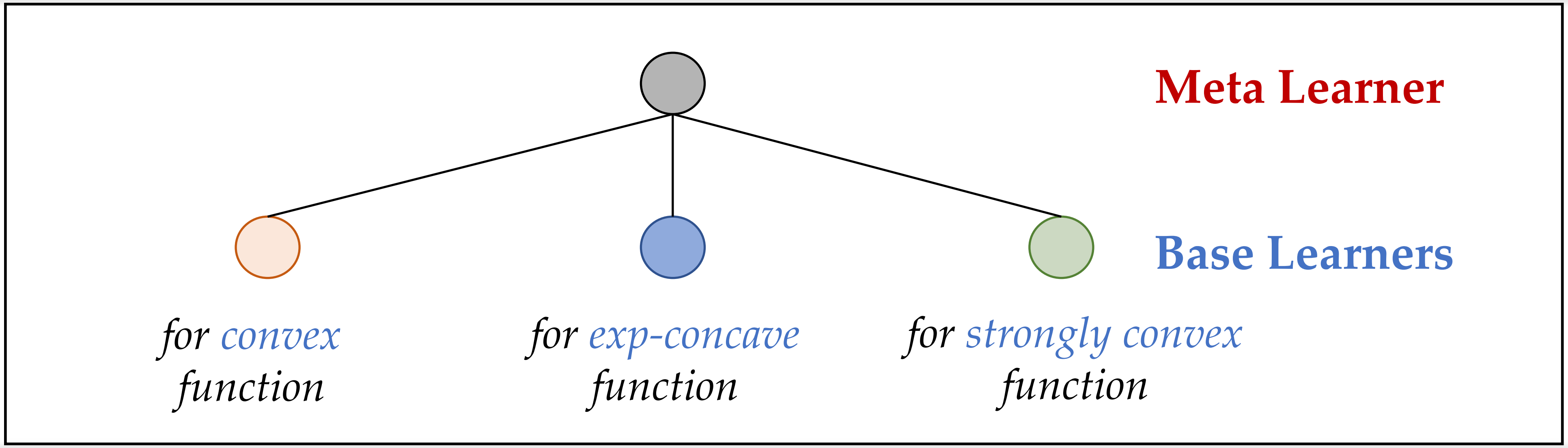
Approximately $V_T \approx \sigma_{1:T}^2 + \Sigma_{1:T}^2$. For stochastic optimization, $\sigma_{1:T}^2 = \sigma^2 T$ and $\Sigma_{1:T}^2 = 0$. For adversarial optimization, $\sigma_{1:T}^2 = 0$ and $\Sigma_{1:T}^2 = V_T$.

Outline

- Background
- Motivation
- Our Approach
- Conclusion

Universal Online Learning

- **Basic idea: Online Ensemble** $\mathbf{x}_t = \sum_{i=1}^{N} p_{t,i}\mathbf{x}_{t,i}$
  - $\boldsymbol{p}_t = [p_{t,1}, \ldots, p_{t,N}]^\top$ is the meta weight;
  - $\{\mathbf{x}_{t,i}\}_{t=1}^{T}$ is the base decisions of the i-th base learners, $i \in [N]$.



also used in non-stationary online learning (for dynamic/adaptive regret minimization)

Universal Online Learning

- **Regret decomposition:** how to control meta-regret in two layers

$$\textsc{Reg}_T = \underbrace{\left[\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_{t,i^\star})\right]}_{\textit{meta regret}} + \underbrace{\left[\sum_{t=1}^{T} f_t(\mathbf{x}_{t,i^\star}) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})\right]}_{\textit{base regret}}$$

- **Key idea:** exploiting the ***second-order regret bound*** on the meta level

[Zhang et al., ICML'22]

$$\sum_{t=1}^{T} \langle p_t, \ell_t \rangle - \sum_{t=1}^{T} \ell_{t,i} \leq \mathcal{O}\left(\sqrt{\sum_{t=1}^{T} r_{t,i}^2}\right)$$

(second-order bound, e.g., Adapt-ML-Prod) [Gaillard et al, COLT'14]

$\Rightarrow$ $\ell_{t,i} \triangleq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_{t,i} \rangle$, $r_{t,i} \triangleq \langle p_t, \ell_t \rangle - \ell_{t,i}$

$$\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star} \rangle \lesssim \sqrt{\sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star} \rangle^2}$$

Universal Online Learning

- **Regret decomposition:** how to control meta-regret in two layers

$$\textsc{Reg}_T = \underbrace{\left[\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star})\right]}_{\textit{meta regret}} + \underbrace{\left[\sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})\right]}_{\textit{base regret}}$$

- **Key idea:** exploiting the ***second-order regret bound*** on the meta level

$\Rightarrow$

$$\ell_{t,i} \triangleq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_{t,i}\rangle$$
$$r_{t,i} \triangleq \langle p_t, \ell_t\rangle - \ell_{t,i}$$

$$\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle \lesssim \sqrt{\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle^2}$$

e.g., **exp-concave**

$\Rightarrow$

$$\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) \leq \sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle - \sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle^2 \leq \mathcal{O}(1)$$

Universal Online Learning

- **Regret decomposition:** how to control meta-regret in two layers

$$\mathrm{REG}_T = \left[\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star})\right] + \left[\sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})\right]$$

meta regret *base regret*

- **Key idea:** exploiting the ***second-order regret bound*** on the meta level

$$\Rightarrow \quad \ell_{t,i} \triangleq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_{t,i}\rangle \qquad r_{t,i} \triangleq \langle p_t, \ell_t\rangle - \ell_{t,i} \qquad \sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle \lesssim \sqrt{\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle^2}$$

e.g., **strongly convex**

$$\Rightarrow \quad \sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) \le \sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle - \sum_{t=1}^T \|\mathbf{x}_t - \mathbf{x}_{t,i^\star}\|^2 \le \mathcal{O}(1)$$

Universal Online Learning

- **Regret decomposition:** how to control meta-regret in two layers

$$\textsc{Reg}_T = \left[\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star})\right] + \left[\sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})\right]$$

meta regret *base regret*

- **Key idea:** exploiting the ***second-order regret bound*** on the meta level

$\Rightarrow$ $\ell_{t,i} \triangleq \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_{t,i}\rangle$, $r_{t,i} \triangleq \langle p_t, \ell_t\rangle - \ell_{t,i}$

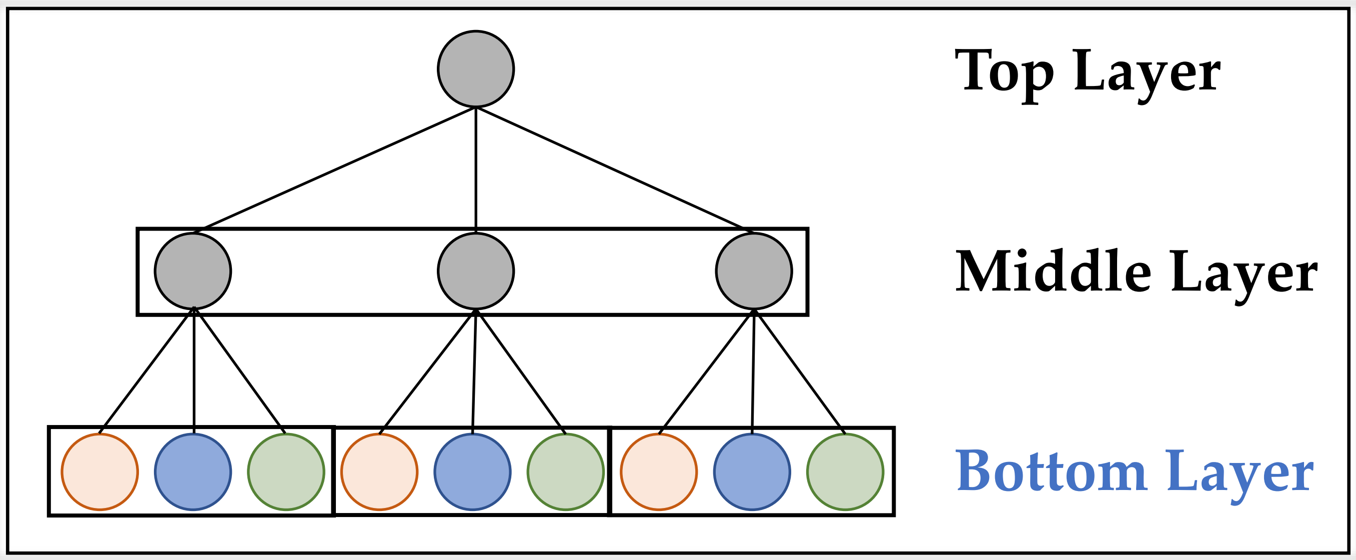
$$\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle \lesssim \sqrt{\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle^2}$$

e.g., **convex**

$\Rightarrow$
$$\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) \leq \sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle \lesssim \sqrt{\sum_{t=1}^T \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star}\rangle^2}$$

Our Approach

- **Multi-layer Online Ensemble**



➢ **Top layer & Middle layer:**

a ***two-layer meta learner***

➢ **Bottom layer:**

basic online ensemble idea

Why three layers? (mostly due to the technical reasons)

Technically, this is due to the *simultaneous requirements of second-order bound (for universality) and negative terms (for gradient variation)*. So we have to use a two-layer online algorithm (MsMwC over MsMwC) [Chen-Wei-Luo, COLT'21] as the meta-learner.

Key Ingredients

- **Ingredient I:** novel ***optimism*** to reuse historical gradients ***universally***

To obtain gradient-variation bounds, we need to reuse historical data, i.e., optimistic online learning.

Recall meta regret: $\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_{t,i^\star})$

we optimize the linearized regret: $\sum_{t=1}^{T} r_{t,i^\star} \triangleq \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star} \rangle$

Optimistic-Adapt-ML-Prod: $$\sum_{t=1}^{T} r_{t,i^\star} \leq \mathcal{O}\left(\sqrt{\sum_{t=1}^{T} (r_{t,i^\star} - m_{t,i^\star})^2}\right)$$

[Wei et al., NIPS'16]

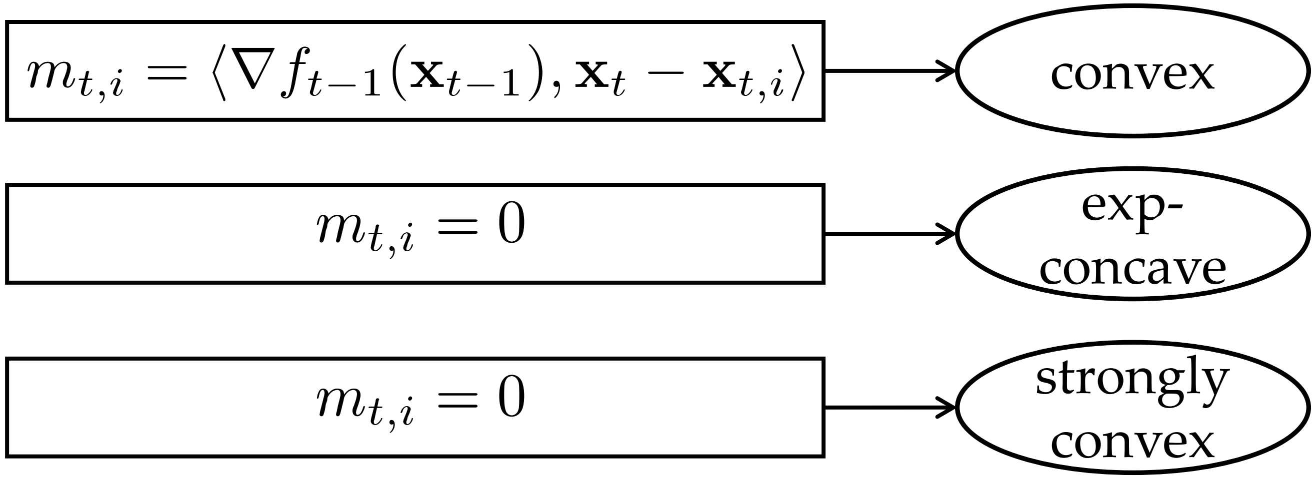
optimism

Key Ingredients

- **Ingredient I:** novel ***optimism*** to reuse historical gradients ***universally***

Goal: to ensure an $\mathcal{O}(1)$ meta regret for ***exp-concave/strongly convex*** functions, and $\mathcal{O}(\sqrt{V_T})$ meta regret for ***convex*** functions.

Challenge: can only use ***separate*** parameters to act as the optimism



different parameters for different functions (not universal)

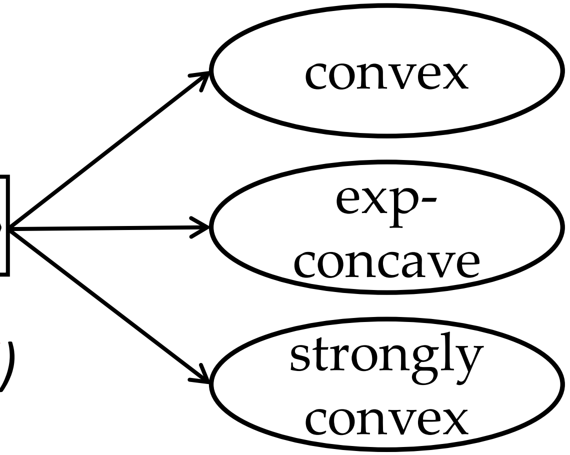
Key Ingredients

- **Ingredient I:** novel ***optimism*** to reuse historical gradients ***universally***

Our solution:

universal parameter

$$m_{t,i} = r_{t-1,i} = \langle \nabla f_{t-1}(\mathbf{x}_{t-1}), \mathbf{x}_{t-1} - \mathbf{x}_{t-1,i} \rangle$$



one parameter for different functions (universal)

$$\sum_{t=1}^{T} (r_{t,i^\star} - m_{t,i^\star})^2 \le \begin{cases} \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}_{t,i^\star} \rangle^2, & \text{(exp-concave \& strongly convex)} \\ V_T + \sum_{t=2}^{T} \|\mathbf{x}_{t,i^\star} - \mathbf{x}_{t-1,i^\star}\|^2 + \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2. & \text{(convex)} \end{cases}$$

algorithm stability

Key Ingredients

- **Ingredient II:** ***collaboration*** in multiple layers to handle the ***stability***

Goal: to ensure the stability $\sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2$ can be handled by the negative regret within the dynamics of online ensemble.

Two layers:

[Zhao et al, 2021]

$$\underbrace{\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2 \lesssim \|\boldsymbol{p}_t - \boldsymbol{p}_{t-1}\|_1^2}_{\text{meta stability}} + \underbrace{\sum_{i=1}^{N} p_{t,i} \|\mathbf{x}_{t,i} - \mathbf{x}_{t-1,i}\|_2^2}_{\text{weighted combination of base stability}}$$

- meta stability: handled by negative terms in meta regret
- weighted stability: collaboration among layers, penalizing unstable base learners

$$\sum_{t=1}^{T} \langle \boldsymbol{\ell}_t + \boldsymbol{b}_t, \boldsymbol{p}_t - \boldsymbol{e}_{i^\star} \rangle \leq X \Longrightarrow \sum_{t=1}^{T} \langle \boldsymbol{\ell}_t, \boldsymbol{p}_t - \boldsymbol{e}_{i^\star} \rangle \leq X - \sum_{t=1}^{T} \sum_{i=1}^{N} p_{t,i} b_{t,i} + \sum_{t=1}^{T} b_{t,i^\star}$$

Key Ingredients

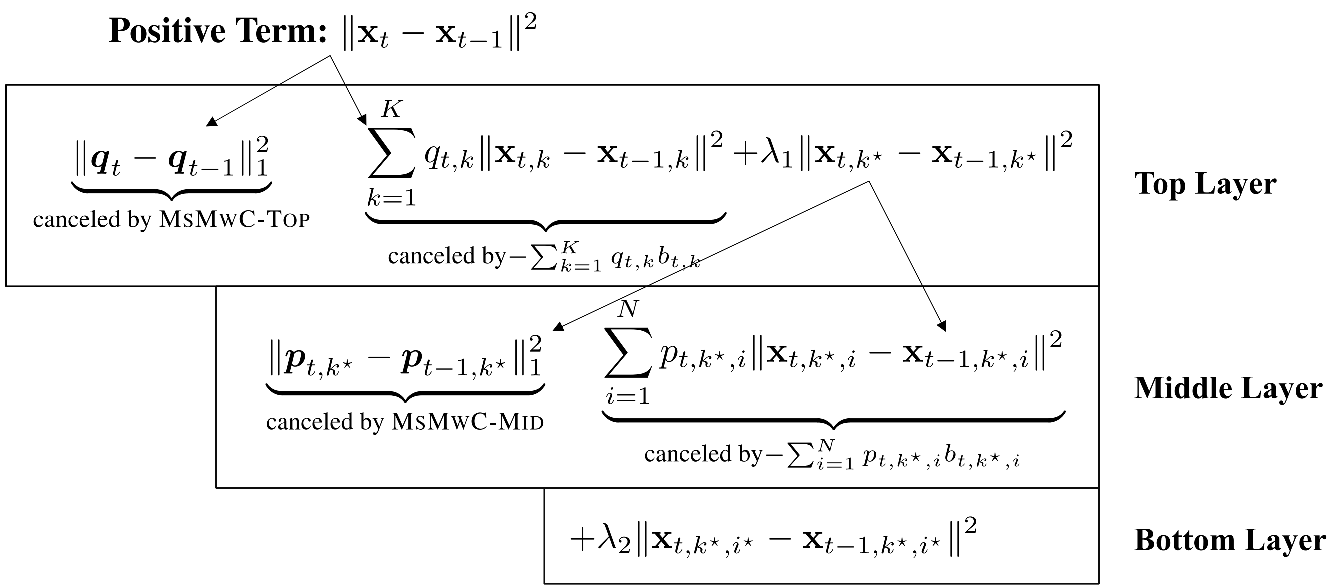
- **Ingredient II:** ***collaboration*** in multiple layers to handle the ***stability***

Goal: to ensure the stability $\sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2$ can be handled by the negative regret within the dynamics of online ensemble.

Three layers:

$$\mathbf{x}_t = \sum_{k=1}^{K} q_{t,k} \mathbf{x}_{t,k}$$

$$\mathbf{x}_{t,k} = \sum_{i=1}^{N} p_{t,k,i} \mathbf{x}_{t,k,i}$$



A principled way to control algorithmic stability in multi-layer structures.

Algorithm

Algorithm 1 Universal OCO with Gradient-variation Guarantees

Input: Curvature coefficient pool $\mathcal{H}$, MsMwC-Mid number K, base learner number N

Initialize: Top layer: $\mathcal{A}^{\text{top}}$ — MsMwC-Top with $\eta_k = (C_0 \cdot 2^k)^{-1}$ and $\widehat{q}_{1,k} = \eta_k^2 / \sum_{k=1}^K \eta_k^2$
Middle layer: $\{\mathcal{A}_k^{\text{mid}}\}_{k \in [K]}$ — MsMwC-Mid with step size $2\eta_k$ and $\widehat{p}_{1,k,i} = 1/N$
Bottom layer: $\{\mathcal{B}_{k,i}\}_{k \in [K], i \in [N]}$ — base learners as specified in Section 2

for $t = 1$ **to** T **do**
    Receive $\mathbf{x}_{t,k,i}$ from $\mathcal{B}_{k,i}$, obtain $\mathbf{x}_{t,k} = \sum_i p_{t,k,i} \mathbf{x}_{t,k,i}$ and submit $\mathbf{x}_t = \sum_k q_{t,k} \mathbf{x}_{t,k}$
    Suffer $f_t(\mathbf{x}_t)$ and observe the gradient information $\nabla f_t(\cdot)$
    Construct $(\boldsymbol{\ell}_t, \boldsymbol{m}_t)$ (3.3) for $\mathcal{A}^{\text{top}}$ and $(\boldsymbol{\ell}_{t,k}, \boldsymbol{m}_{t,k})$ (3.4) for $\mathcal{A}_k^{\text{mid}}$
    $\mathcal{A}^{\text{top}}$ updates to $\boldsymbol{q}_{t+1}$ and $\mathcal{A}_k^{\text{mid}}$ updates to $\boldsymbol{p}_{t+1,k}$
    Multi-Gradient Feedback Model:
        Send gradient $\nabla f_t(\cdot)$ to $\mathcal{B}_{k,i}$ for update ▷ $\mathcal{O}(\log^2 T)$ gradient queries
    One-Gradient Feedback Model:
        Construct surrogates $h_{t,i}^{\text{sc}}(\cdot)$, $h_{t,i}^{\text{exp}}(\cdot)$, $h_{t,i}^{\text{c}}(\mathbf{x})$ using only $\nabla f_t(\mathbf{x}_t)$
        Send the surrogate functions to $\mathcal{B}_{k,i}$ for update ▷ Only *one* gradient query
end for

Main Result

- The first ***universal*** algorithm with near-optimal ***gradient-variation regret***.

Theorem 1 (Yan-Z-Zhou; NeurIPS 2023). *Under standard assumptions, our algorithm enjoys*

- *it achieves* $\mathcal{O}(\log V_T)$ *regret for strongly convex functions;*
- *it achieves* $\mathcal{O}(d \log V_T)$ *regret for exp-concave functions;*
- *it achieves* $\widehat{\mathcal{O}}(\sqrt{V_T})$ *regret for convex functions.*

Here, $V_T = \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla f_t(\mathbf{x}) - \nabla f_{t-1}(\mathbf{x})\|^2$ *is gradient variation and* $\widehat{\mathcal{O}}(\cdot)$ *omits* $\log V_T$ *factors.*

Immediate implications to game theory and SEA model.

Result for SEA

- **Stochastically Extended Adversarial (SEA)** [Sachs et al., NeurIPS'22]

Interpolation between stochastic and adversarial online convex optimization

Two crucial complexity measures:

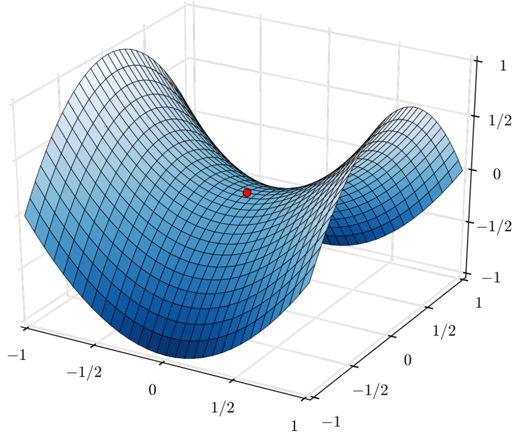
$$\sigma_{1:T}^2 \triangleq \sum_{t=1}^{T} \max_{\mathbf{x}\in\mathcal{X}} \mathbb{E}_{f_t\sim\mathcal{D}_t}[\|\nabla f_t(\mathbf{x}) - \nabla F_t(\mathbf{x})\|^2], \quad \Sigma_{1:T}^2 \triangleq \mathbb{E}\left[\sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \|\nabla F_t(\mathbf{x}) - \nabla F_{t-1}(\mathbf{x})\|^2\right]$$

stochastic change *adversarial change*

Theorem 2. *Under standard assumptions, our algorithm obtains* $\mathcal{O}((\sigma_{\max}^2 + \Sigma_{\max}^2)\log(\sigma_{1:T}^2 + \Sigma_{1:T}^2))$ *regret for strongly convex functions,* $\mathcal{O}(d\log(\sigma_{1:T}^2 + \Sigma_{1:T}^2))$ *regret for exp-concave functions and* $\widehat{\mathcal{O}}(\sqrt{(\sigma_{1:T}^2 + \Sigma_{1:T}^2)})$ *regret for convex functions.*

Result for Games

- **Min-Max Optimization**



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

Consider two aspects:

(i) curvatures: f is bilinear/strongly convex-concave

(ii) honest: all players run the same algo; dishonest: otherwise (some may disobey)

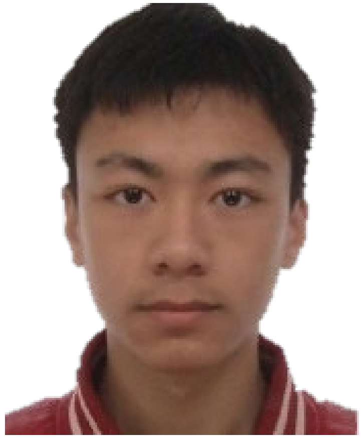
Theorem 3. *Under standard assumptions, for bilinear and strongly convex-concave games, our algorithm enjoys $\mathcal{O}(1)$ regret summation in the honest case, $\widehat{\mathcal{O}}(\sqrt{T})$ and $\mathcal{O}(\log T)$ bounds respectively in the dishonest case.*

Conclusion

- Consider two-level adaptivity for online convex optimization.
- **Universal online learning with Gradient-Variation Regret**
  - deploying a single algorithm to achieve multiple (near-)optimal guarantees for different function families
  - using multi-layer online ensemble with carefully designed optimism and corrections to achieve the desired gradient-variation regret
  - Gradient-variation regret is useful for game theory and stochastic opt.
- ***Open problems***
  - Is the three-layer ensembling structure necessary?
  - How to achieve the strictly optimal result for convex functions?
  - How to extend to more challenging adaptive/dynamic regret minimization?

Collaborators

Yu-Hu Yan (NJU)



Yu-Jie Zhang (U Tokyo)

Lijun Zhang (NJU)

Zhi-Hua Zhou (NJU)

Yu-Jie is actively finding the postdoc opportunity!

Thanks!

References

- Y.-H. Yan, P. Zhao, and Z.-H. Zhou. Universal online learning with gradient variations: A multi-layer online ensemble approach. NeurIPS 2023. **(major ref)**
- P. Zhao, Y.-J. Zhang, L. Zhang, and Z.-H. Zhou. Adaptivity and non-stationarity: Problem-dependent dynamic regret for online convex optimization. ArXiv preprint: 2112.14368, 2021.
- L. Zhang, G. Wang, J. Yi, and T. Yang. A simple yet universal strategy for online convex optimization. ICML 2022.
- L. Chen, H. Luo, and C. Wei. Impossible tuning made possible: A new expert algorithm and its applications. COLT 2021.

Thanks!

References

- V. Syrgkanis, A. Agarwal, H. Luo, and R. E. Schapire. Fast convergence of regularized learning in games. NIPS 2015.
- P. Gaillard, G. Stoltz, and T. van Erven. A second-order bound with excess losses. COLT 2014.
- S. Sachs, H. Hadiji, T. van Erven, and C. Guzmán. Between stochastic and adversarial online convex optimization: Improved regret bounds via smoothness. NeurIPS 2022.
- C. Wei, Y. Hong, and C. Lu. Tracking the best expert in non-stationary stochastic environments. NIPS 2016.

Thanks!