Universal Online Learning with Gradient Variations

Yu-Hu Yan

Peng Zhao

Zhi-Hua Zhou

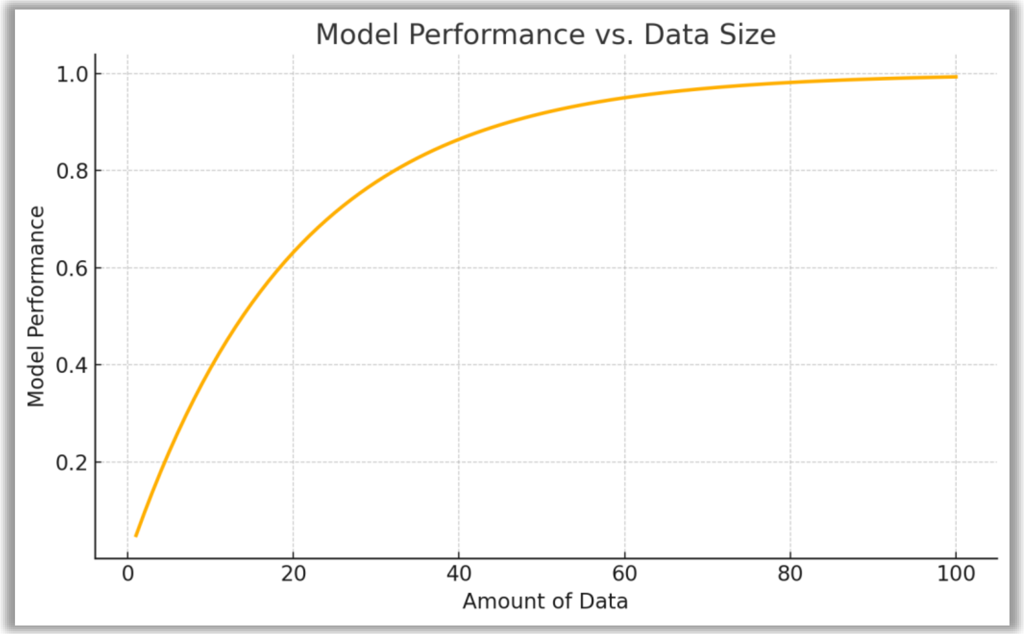
Presented by Yu-Hu Yan

2025.04.18

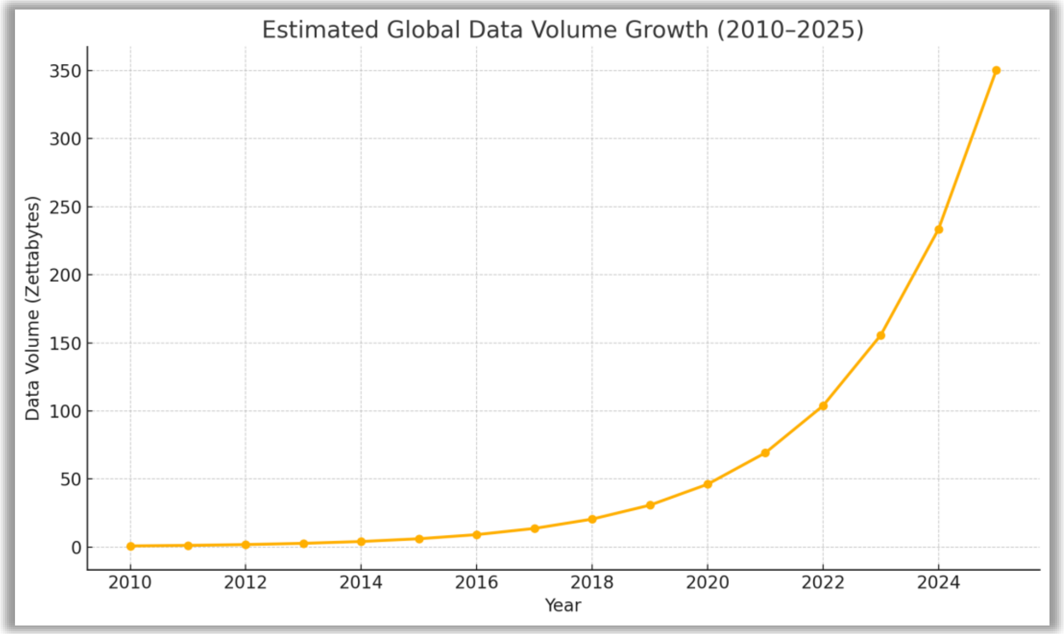
Universal online learning with gradient variations: A multi-layer online ensemble approach, NeurIPS 2023 (**Spotlight**)
A simple and optimal approach for universal online learning with gradient variations, NeurIPS 2024

Big Data Era

- Machine learning ***needs and has to*** handle ***big data***



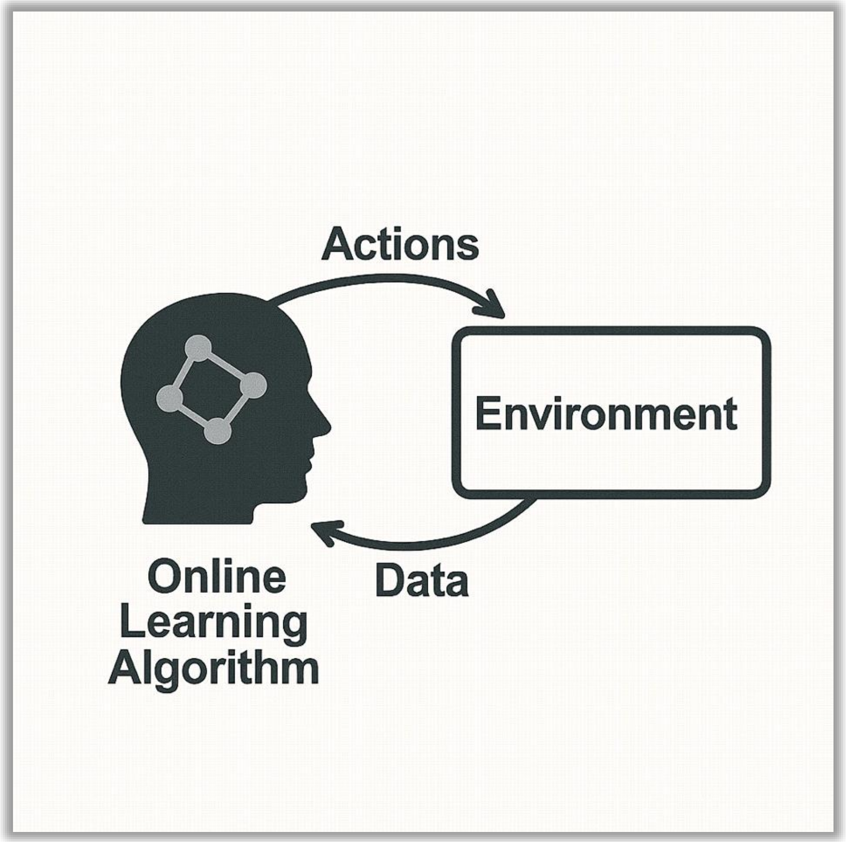
More data generally leads to better performance—up to a point.



The global data volume is growing exponentially.

Big Data Era

- Why ***online learning*** is essential in the era of big data



- **Big data arrives continuously.**

  Data isn't static—it streams in every second.

- **Retraining from scratch is inefficient.**

  Batch learning can't keep up with real-time needs.

- **Online learning enables real-time updates.**

  Models adapt on the fly with minimal delay.

Formalization

Online Learning

At each round $t = 1, 2 \cdots, T$

1. the learner first pick a point $\mathbf{w}_t \in \mathcal{W}$;
2. and simutaneously the environment picks an online function $f_t : \mathcal{W} \mapsto [0,1]$ to evaluate the model;
3. the learner then suffers loss $f_t(\mathbf{w}_t)$ and observes some information of f_t.

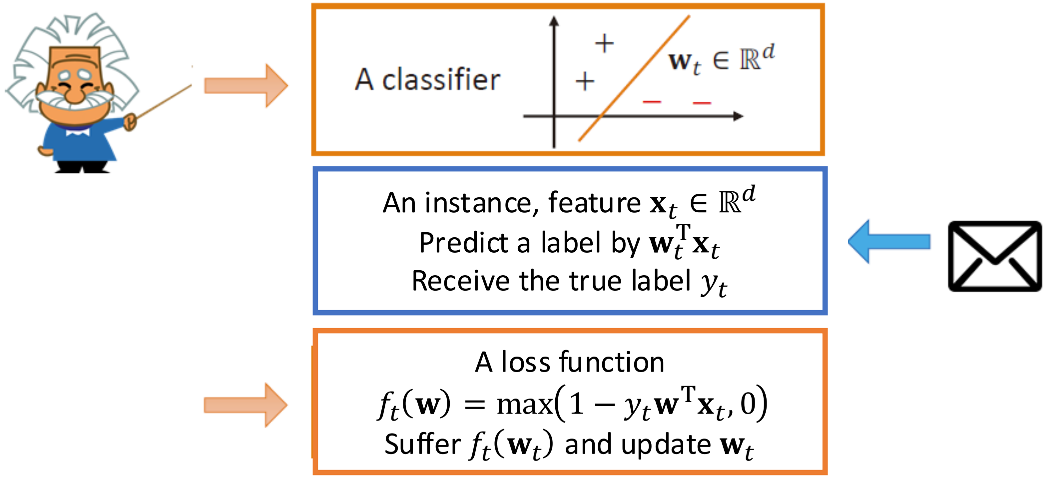
Example: online function $f_t : \mathcal{W} \mapsto \mathbb{R}$ is composition of

(i) the loss $\ell : \hat{\mathcal{Y}} \times \mathcal{Y} \mapsto \mathbb{R}$, and

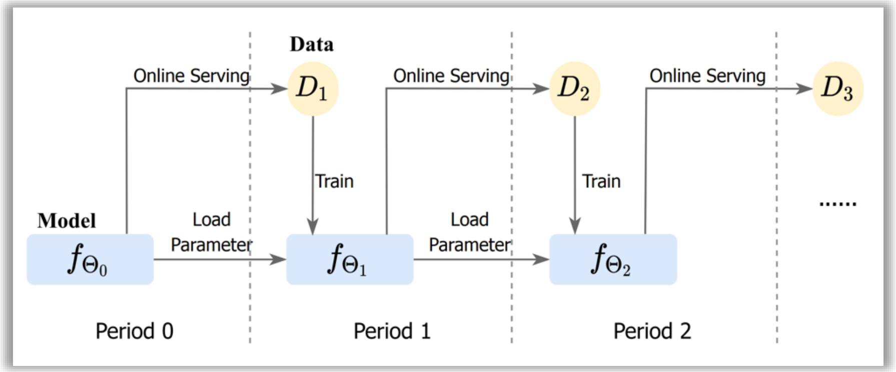
(ii) the hypothesis function $h : \mathcal{W} \times \mathcal{X} \mapsto \hat{\mathcal{Y}}$.

$$\Rightarrow f_t(\mathbf{w}) = \ell(h(\mathbf{w}; \mathbf{x}_t), y_t) = \ell(\mathbf{w}^\top \mathbf{x}_t, y_t)$$

Example I: Spam filtering



Example II: Online recommandation



Formalization

Learning And Mining from DatA

Online Convex Optimization (OCO)

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

- the learner submits an online model $\mathbf{x}_t \in \mathcal{X} \subseteq \mathbb{R}^d$
- simutaneously, environments decide a convex loss function $f_t : \mathcal{X} \mapsto \mathbb{R}$
- the learner suffers $f_t(\mathbf{x}_t)$ and receives gradient info. of the loss function

Regret: Online prediction as good as the best offline model

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

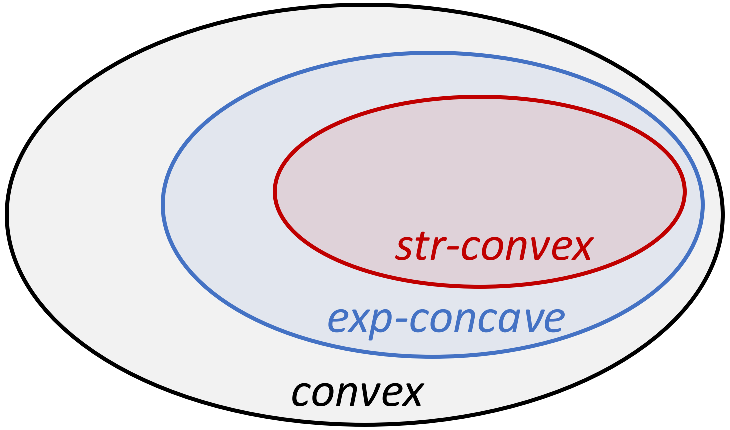
cumulative loss of best offline model

cumulative loss of the online model

Problem Setup

☐ Universal Online Learning

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



Online learning usually considers **three** kinds of curvatures:

- ***convex***: $f(\mathbf{x}) - f(\mathbf{y}) \le \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$.
- ***λ-strongly convex***: $f(\mathbf{x}) - f(\mathbf{y}) \le \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle - \frac{\lambda}{2}\|\mathbf{x} - \mathbf{y}\|^2$.
- ***α-exp-concave***: $f(\mathbf{x}) - f(\mathbf{y}) \le \langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle - \frac{\alpha}{2}\langle \nabla f(\mathbf{x}), \mathbf{x} - \mathbf{y}\rangle^2$.

Problem Setup

Universal Online Learning

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

In OCO, the type of ***functional curvature*** plays an important role in the best attainable regret bounds.

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 $\eta_t = \frac{1}{\lambda t}$	$\mathcal{O}(\frac{1}{\lambda} \cdot \log T)$
α-exp-concave	Online Newton Step with α	$\mathcal{O}(\frac{1}{\alpha} \cdot d \log T)$

Problem Setup

Universal Online Learning

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

In OCO, the type of ***functional curvature*** plays an important role in the best attainable regret bounds.

Common algorithm is only suitable for ***one specific curvature type***.

What if the ***curvature type (and coefficient) is unknown***?

In this talk, we focus on ***universal online learning***, where the curvature is unknown.

Problem Setup

LAMDA Learning And Mining from DatA

Universal Online Learning

$$\textsc{Reg}_T(\mathcal{A}, \{f_t\}_{t=1}^T) \triangleq \sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^T f_t(\mathbf{x})$$

In this talk, we focus on ***universal online learning***, where the curvature is unknown.

Universal Regret Minimization

$$\textsc{Reg}_T(\mathcal{A}, \{f_t\}_{t=1}^T) = \begin{cases} \textsc{Reg}_T(\mathcal{A}_{\text{sc}}, \mathcal{F}_{\text{sc}}^{\lambda}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{sc}}^{\lambda}, \\ \textsc{Reg}_T(\mathcal{A}_{\text{ec}}, \mathcal{F}_{\text{ec}}^{\alpha}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{ec}}^{\alpha}, \\ \textsc{Reg}_T(\mathcal{A}_{\text{c}}, \mathcal{F}_{\text{c}}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{c}}, \end{cases}$$

Problem Setup

LAMDA Learning And Mining from DatA

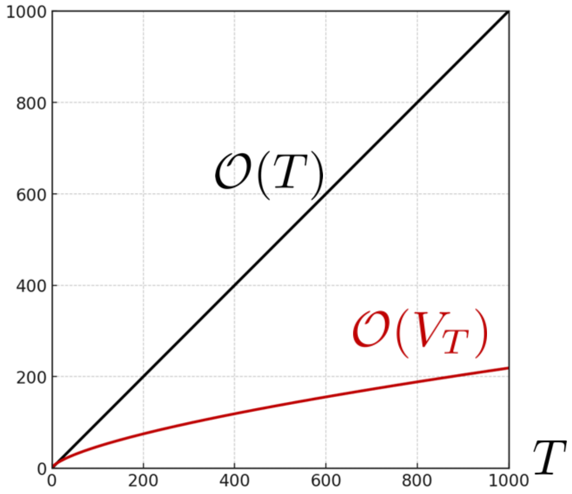
☐ **Problem-dependent** regret

- Regret measured by T only considers the ***worst-case*** scenarios.
- Can we exploit the ***niceness*** of environments for improved result?

Gradient variation:

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

cumulative variations in gradients,
reflecting the difficulty of online problems



The regret bounds can be strengthened to $\mathcal{O}(\frac{1}{\lambda}\log V_T)$, $\mathcal{O}(\frac{d}{\alpha}\log V_T)$, and $\mathcal{O}(\sqrt{V_T})$.

Problem Setup

Gradient variation:

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

cumulative variations in gradients, reflecting the difficulty of online problems.

Why do we study gradient variation?

(i) Gradient variation implies other problem-dependent quantities ***directly in analysis***.

e.g.,

***Small-loss* term:**

$$F_T \triangleq \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$$

cumulative loss of the best model

***Gradient-variance* term:**

$$W_T \triangleq \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} \left\| \nabla f_t(\mathbf{x}) - \frac{1}{T}\sum_{s=1}^{T} \nabla f_s(\mathbf{x}) \right\|^2$$

variance of gradients

(ii) Gradient variation can ***bridge stochastic and adversarial online optimization***.

- [Sachs et al., Between stochastic and adversarial online convex optimization: Improved regret bounds via smoothness, NeurIPS 2022]

(iii) Gradient variation in achieving ***fast rates in games***.

- [Syrgkanis et al., Fast convergence of regularized learning in games, NIPS 2015 (Best Paper Award)]

(iv) Gradient variation in ***accelerated convex smooth optimization***.

A General Framework

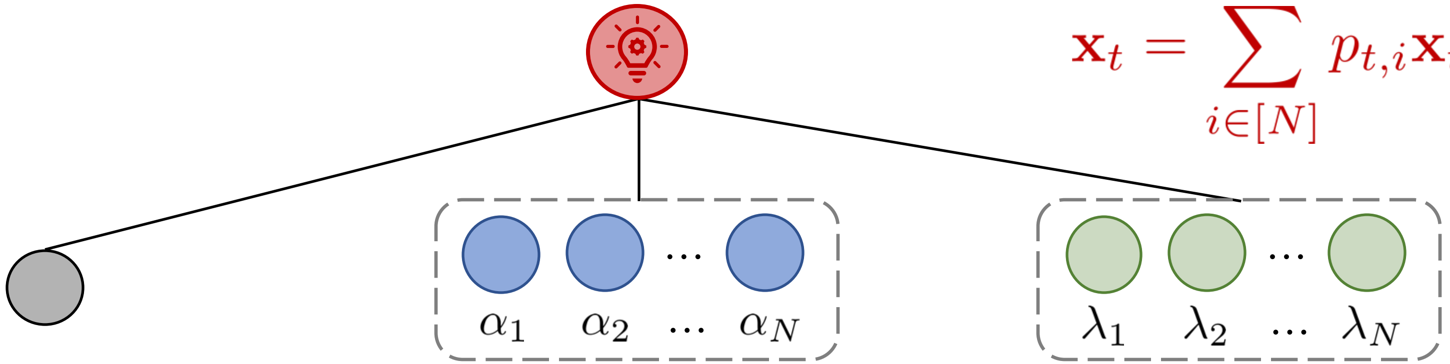
Universal Regret Minimization

$$\textsc{Reg}_T(\mathcal{A}, \{f_t\}_{t=1}^T) = \begin{cases} \textsc{Reg}_T(\mathcal{A}_{\text{sc}}, \mathcal{F}_{\text{sc}}^{\lambda}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{sc}}^{\lambda}, \\ \textsc{Reg}_T(\mathcal{A}_{\text{ec}}, \mathcal{F}_{\text{ec}}^{\alpha}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{ec}}^{\alpha}, \\ \textsc{Reg}_T(\mathcal{A}_{\text{c}}, \mathcal{F}_{\text{c}}), & \text{when } \{f_t\}_{t=1}^T \text{ belongs to } \mathcal{F}_{\text{c}}, \end{cases}$$

◻ Online Ensemble [Zhao-Zhang-Zhang-Zhou, JMLR 2024]

General goal: *To handle the* ***uncertainty*** *of environments.*

$$\mathbf{x}_t = \sum_{i \in [N]} p_{t,i} \mathbf{x}_{t,i}$$ **Meta Learner**



Base Learners

for convex function | *for exp-concave function* | *for strongly convex function*

($\alpha_1, \alpha_2, \ldots, \alpha_N$ are discritized guesses of the true α)

- ➢ **Base learners** guess the curvature (str-convex/exp-concave/cvx).
- ➢ **Meta learner** tracks the best base learner.

Main Result

- Our first work [Yan-Zhao-Zhou, NeurIPS 2023]

*A **single** algorithm with **near-optimal** universal gradient-variation regret.*

Theorem 1. *Under standard assumptions, our algorithm ensures that*

$$\mathrm{REG}_T(\mathcal{A}, \{f_t\}_{t=1}^T) \leq \begin{cases} \mathcal{O}(\frac{1}{\lambda} \cdot \log V_T), & \text{when } \{f_t\}_{t=1}^T \text{ are } \lambda\text{-strongly convex,} \\ \mathcal{O}(\frac{1}{\alpha} \cdot d \log V_T), & \text{when } \{f_t\}_{t=1}^T \text{ are } \alpha\text{-exp-concave,} \\ \mathcal{O}(\sqrt{V_T \log V_T}), & \text{when } \{f_t\}_{t=1}^T \text{ are convex.} \end{cases}$$

Main Result

Learning And Mining from DatA

- Our **second** work [Yan-Zhao-Zhou, NeurIPS 2024]

*A **single** algorithm with **the optimal** universal gradient-variation regret.*

Theorem 2. *Under standard assumptions, our algorithm ensures that*

$$\mathrm{REG}_T(\mathcal{A}, \{f_t\}_{t=1}^T) \le \begin{cases} \mathcal{O}(\frac{1}{\lambda} \cdot \log V_T), & \textit{when } \{f_t\}_{t=1}^T \textit{ are } \lambda\textit{-strongly convex}, \\ \mathcal{O}(\frac{1}{\alpha} \cdot d \log V_T), & \textit{when } \{f_t\}_{t=1}^T \textit{ are } \alpha\textit{-exp-concave}, \\ \mathcal{O}(\sqrt{V_T}), & \textit{when } \{f_t\}_{t=1}^T \textit{ are convex}. \end{cases}$$

Main Result

LAMDA Learning And Mining from DatA

☐ Comparison of two works

Works	Regret Bounds			RVU
	strongly convex	exp-concave	convex	
Our NeurIPS'23	$\log V_T$	$d \log V_T$	$\sqrt{V_T \log V_T}$	✓
Our NeurIPS'24	$\log V_T$	$d \log V_T$	$\sqrt{V_T}$	✗

RVU: Regret bounded by **V**ariation in **U**tilities

Remarks:

- NeurIPS'23 enjoys the **RVU property**, which is essential in *game theory*.
- NeurIPS'24 enjoys the **optimal** theoretical guarantees.

Application-I

Two-Player Zero-Sum Game

Example: 

y-player decision $\mathbf{y}_t = (1/2\ 1/2\ 0)^\top$

x-player decision $\mathbf{x}_t = \begin{pmatrix} 1/3 \\ 1/3 \\ 1/3 \end{pmatrix}$

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)

Application-I

Two-Player Zero-Sum Game

Online Game Protocol (Repeated Play)

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 $A\mathbf{x}_t$

Application-I

- **RVU** property is essential for **fast rates** in games [Syrgkanis et al., NIPS 2015 **(Best Paper)**]

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

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

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

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

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

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

$$\Longrightarrow \mathrm{Reg}_T^x + \mathrm{Reg}_T^y \leq \mathcal{O}(1)$$

Application-I

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:

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

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

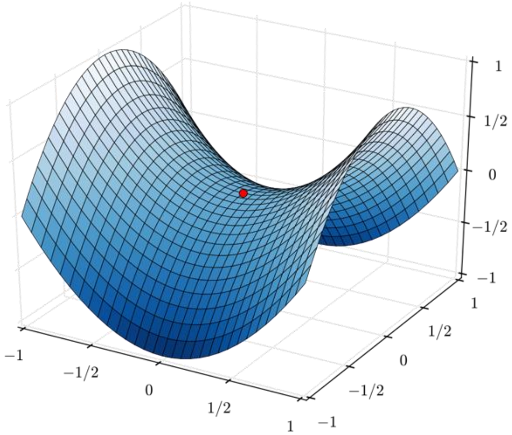
$$\mathrm{Reg}_T^y \lesssim 1 + \underbrace{\sum_{t=2}^{T} \|\mathbf{x}_t^\top A - \mathbf{x}_{t-1}^\top A\|_\infty^2}_{\text{gradient variation}} \underbrace{- \sum_{t=2}^{T} \|\mathbf{y}_t - \mathbf{y}_{t-1}\|_1^2}_{\text{negative stability}}$$

$$\Longrightarrow \mathrm{Reg}_T^x + \mathrm{Reg}_T^y \leq \mathcal{O}(1)$$

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

Application-I

Min-Max Optimization



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

Consider two aspects:

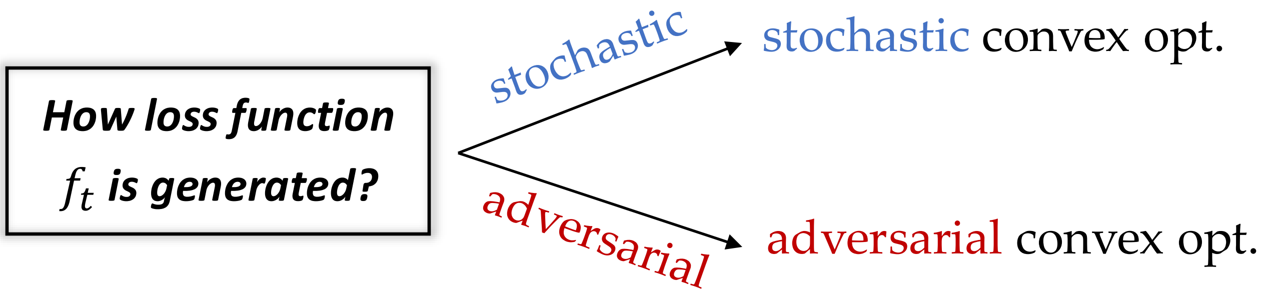
(i) curvatures: f is either **bilinear** or **strongly convex-concave** (but ***unknown***)

(ii) **honest**: all players run the same algo; **dishonest**: someone may disobey

	Bilinear	Strongly-Convex-Strongly-Concave	
Honest	$\mathcal{O}(1)$ reg sum	$\mathcal{O}(1)$ reg sum	*due to gradient variation*
Dishonest	$\mathcal{O}(\sqrt{T\log T})$	$\mathcal{O}(\log T)$	*due to universality*

Application-II

Stochastically Extended Adversarial (SEA) [Sachs et al., NeurIPS 2022]



- Setup: f_t is chosen from a distribution $\mathcal{F}_t$: $f_t \sim \mathcal{F}_t$

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

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

Application-II

☐ Stochastically Extended Adversarial (SEA) [Sachs et al., NeurIPS 2022]

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

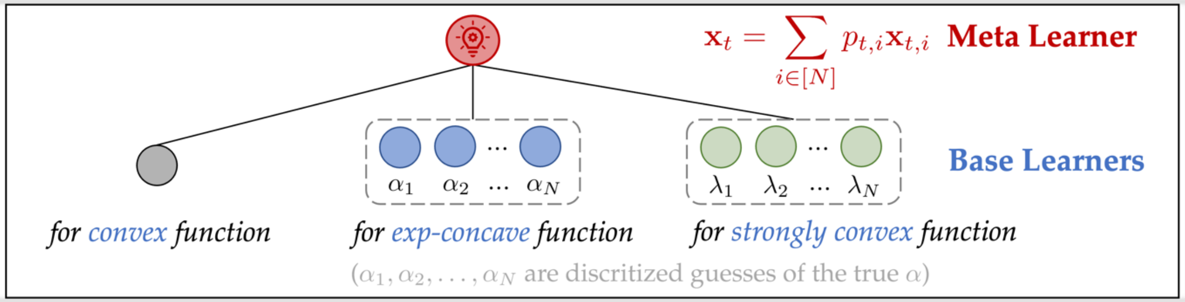
(stochastic variation) *(adversarial variation)*

Theorem 5. *Under boundedness and smoothness of $F_t(\cdot)$ for any $t \in [T]$, our approach ensures*

$$\text{REG}_T \leq \begin{cases} \mathcal{O}\left(\frac{1}{\lambda} \cdot (\sigma_{\max}^2 + \Sigma_{\max}^2) \ln\left(\frac{\sigma_{1:T}^2 + \Sigma_{1:T}^2}{\sigma_{\max}^2 + \Sigma_{\max}^2}\right)\right), & \text{when } \{F_t\}_{t=1}^T \text{ are } \lambda\text{-strongly-cvx}, \\ \mathcal{O}\left(\frac{1}{\alpha} \cdot d \ln(\sigma_{1:T}^2 + \Sigma_{1:T}^2)\right), & \text{when } \{f_t\}_{t=1}^T \text{ are } \alpha\text{-exp-concave}, \\ \mathcal{O}\left(\sqrt{\sigma_{1:T}^2 + \Sigma_{1:T}^2}\right), & \text{when } \{F_t\}_{t=1}^T \text{ are convex}. \end{cases}$$

Matching the ***state-of-the-art*** results attainable when knowing curvature information.

General Analysis



◻ Meta-base regret decomposition:

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

meta regret *base regret*

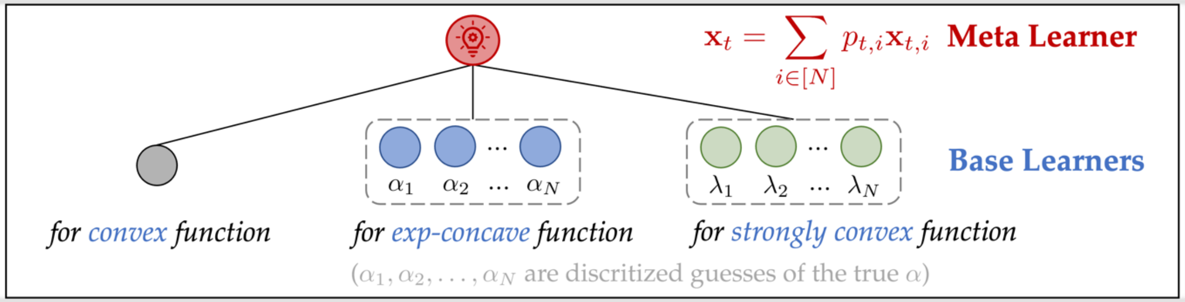
$i^\star$ represents the *best* base learner (right guess of curvature type and coefficient)

Intuitively,

- **Base regret** measures the regret of the best base learner (the *best achievable* result).
- **Meta regret** measures the algorithm's ability to ***track the best base learner***.

*P.S.: Meta learner does **not** need to know which base learner is the best.*

General Analysis



- **Meta-base regret decomposition:**

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

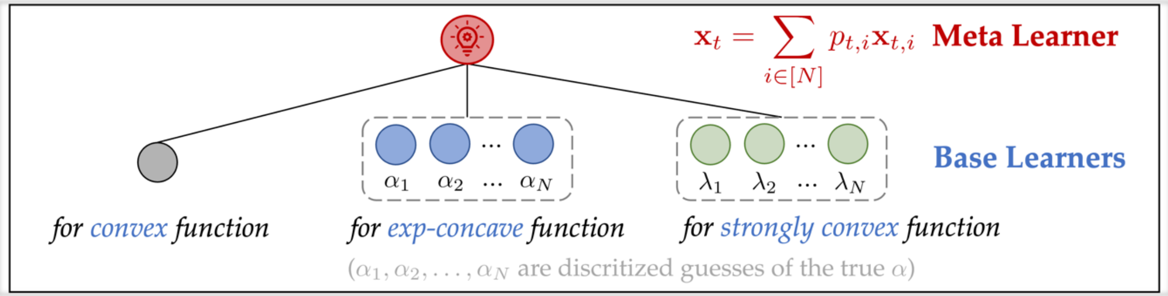
meta regret *base regret*

Optimizing ***meta regret*** as **P**rediction with **E**xpert **A**dvice **(PEA)** problem:

$$\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) = \sum_{t=1}^T f_t\left(\sum_{i=1}^N p_{t,i}\mathbf{x}_{t,i}\right) - \sum_{t=1}^T f_t(\mathbf{x}_{t,i^\star}) \le \sum_{t=1}^T \sum_{i=1}^N p_{t,i} f_t(\mathbf{x}_{t,i}) - f_t(\mathbf{x}_{t,i^\star})$$

$$= \sum_{t=1}^T \langle \boldsymbol{\ell}_t, \boldsymbol{p}_t\rangle - \ell_{t,i^\star} \qquad \text{by defining } \ell_{t,i} \triangleq f_t(\mathbf{x}_{t,i})$$

Technical Challenge



- How to obtain gradient-variation regret?

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

What we have: $\bar{V}_T = \sum_{t=2}^T \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2$

(in the t-th round, we query the gradient of $\nabla f_t(\mathbf{x}_t)$)

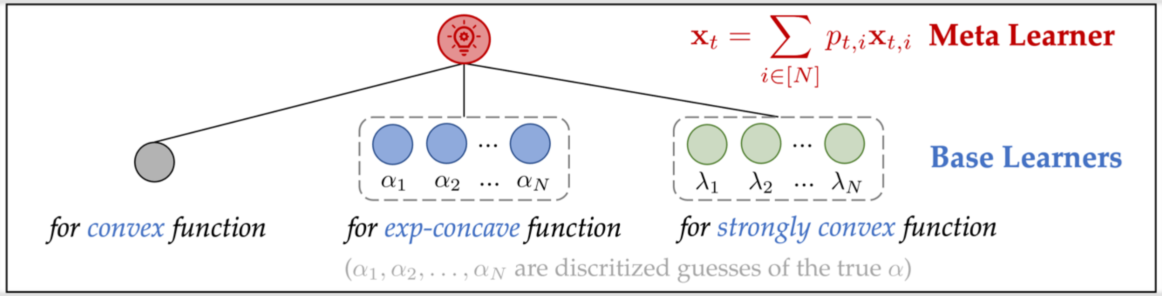
Two technical routines - I:

(smoothness assumption is required)

$$\bar{V}_T \lesssim \sum_{t=2}^T \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_t)\|^2 + \sum_{t=2}^T \|\nabla f_{t-1}(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2 \lesssim V_T + L^2 \sum_{t=2}^T \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2$$

Challenge: $\mathbf{x}_t = \sum_{i\le N} p_{t,i}\mathbf{x}_{t,i}$ is a weighted combination. Thus controlling $\mathbf{x}_t - \mathbf{x}_{t-1}$ requires the ***collaboration*** of meta and base learners.

Technical Challenge



- How to obtain gradient-variation regret?

Two technical routines - I:

$$\bar{V}_T \lesssim \sum_{t=2}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_t)\|^2 + \sum_{t=2}^{T} \|\nabla f_{t-1}(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2 \lesssim V_T + L^2 \sum_{t=2}^{T} \|\mathbf{x}_t - \mathbf{x}_{t-1}\|^2$$

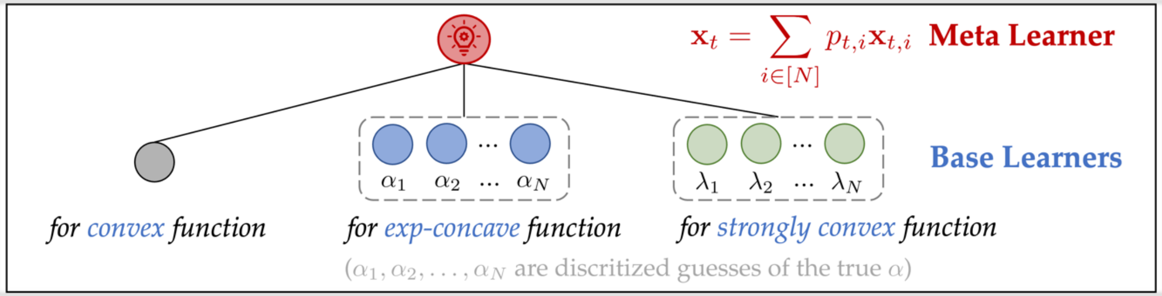
Challenge: $\mathbf{x}_t = \sum_{i\le N} p_{t,i}\mathbf{x}_{t,i}$ is a weighted combination. Thus controlling $\mathbf{x}_t - \mathbf{x}_{t-1}$ requires the ***collaboration*** of meta and base learners.

Decomposition: [Zhao et al., JMLR 2024]

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

Controlling the weighted stability requires ***meta-base collaboration****.*

Technical Challenge



□ How to obtain gradient-variation regret?

Collaborative online ensemble: [Zhao et al., JMLR 2024]

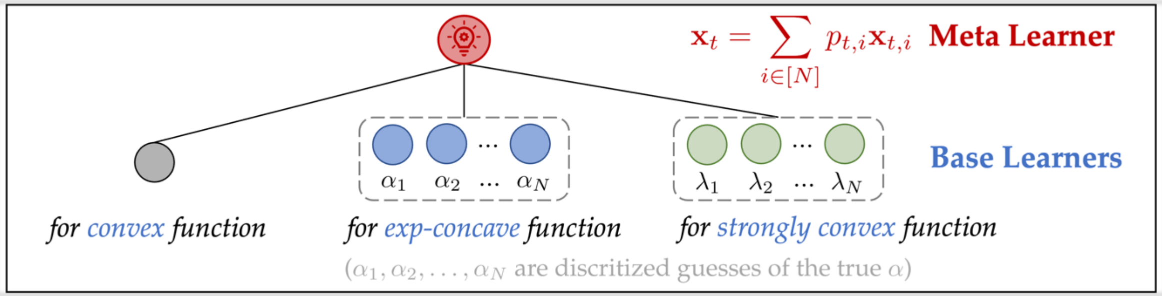
$$\|\mathbf{x}_t - \mathbf{x}_{t-1}\|_2^2 \lesssim \underbrace{\|\boldsymbol{p}_t - \boldsymbol{p}_{t-1}\|_1^2}_{\text{meta stability}} + \underbrace{\sum_{i\le N} p_{t,i}\|\mathbf{x}_{t,i} - \mathbf{x}_{t-1,i}\|_2^2}_{\text{weighted combination of base stability}}$$

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

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

Intuition: Add *corrections* in the meta loss to punish less stable base learners.

Technical Challenge



☐ How to obtain gradient-variation regret?

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

What we have: $\bar{V}_T = \sum_{t=2}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2$

(in the t-th round, we query the gradient of $\nabla f_t(\mathbf{x}_t)$)

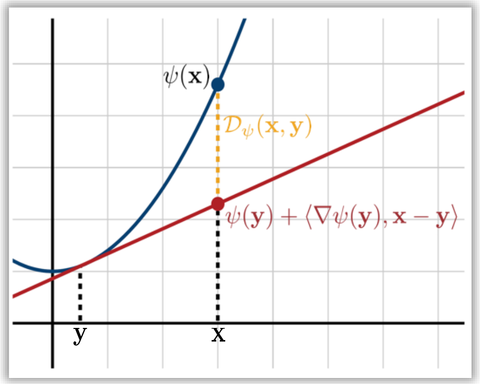
Two technical routines - II:

A ***tighter*** upper bound for ***squared*** gradient change:

Definition 1 (Theorem 2.1.5 of (Nesterov, 2018)). $f(\cdot)$ is L-smooth over $\mathbb{R}^d$ if and only if $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \le 2L\mathcal{D}_f(\mathbf{y},\mathbf{x})$ for any $\mathbf{x},\mathbf{y} \in \mathbb{R}^d$.

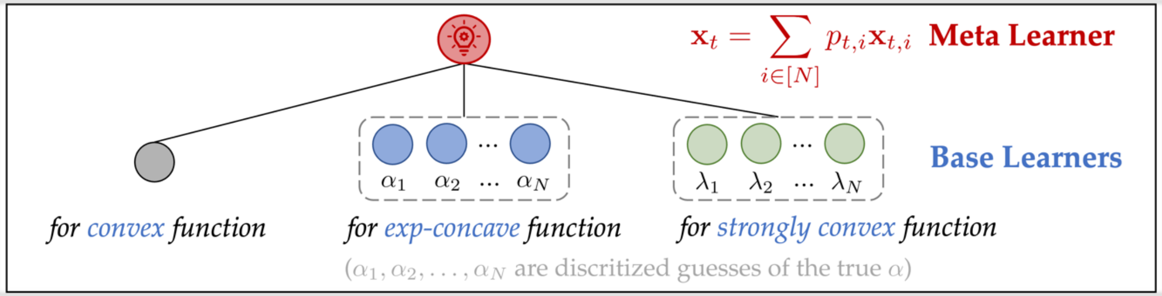
Bregman divergence: $\mathcal{D}_f(\mathbf{x},\mathbf{y}) \triangleq f(\mathbf{x}) - f(\mathbf{y}) - \langle \nabla f(\mathbf{y}), \mathbf{x}-\mathbf{y}\rangle$

tighter than $\|\nabla f_t(\mathbf{x}) - \nabla f_t(\mathbf{y})\|^2 \le L^2\|\mathbf{x}-\mathbf{y}\|^2$ by the smoothness assumption



Bregman divergence

Technical Challenge



- How to obtain gradient-variation regret?

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

What we have: $\bar{V}_T = \sum_{t=2}^T \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2$

(in the t-th round, we query the gradient of $\nabla f_t(\mathbf{x}_t)$)

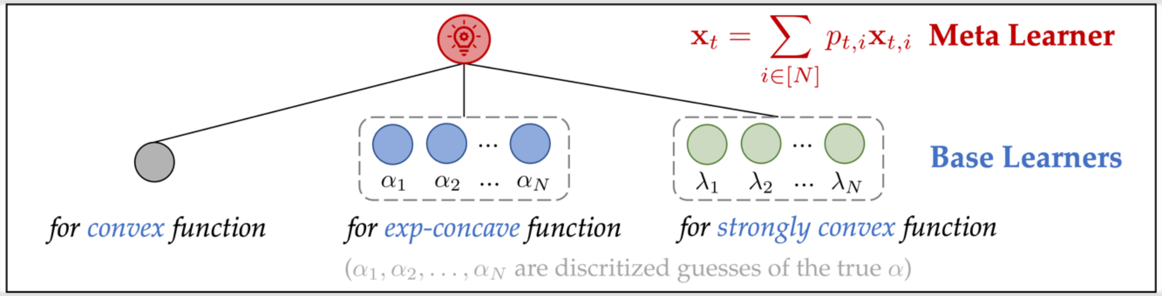
Two technical routines - II:

Definition 1 (Theorem 2.1.5 of (Nesterov, 2018)). $f(\cdot)$ is L-smooth over $\mathbb{R}^d$ if and only if $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \le 2L\mathcal{D}_f(\mathbf{y}, \mathbf{x})$ for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$.

$$\bar{V}_T \lesssim \sum_{t=2}^T \left(\|\nabla f_t(\mathbf{x}_t) - \nabla f_t(\mathbf{x}^\star)\|^2 + \|\nabla f_t(\mathbf{x}^\star) - \nabla f_{t-1}(\mathbf{x}^\star)\|^2 + \|\nabla f_{t-1}(\mathbf{x}^\star) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2\right)$$

$$\lesssim L\sum_{t=2}^T \mathcal{D}_{f_t}(\mathbf{x}^\star, \mathbf{x}_t) + V_T + L\sum_{t=2}^T \mathcal{D}_{f_{t-1}}(\mathbf{x}^\star, \mathbf{x}_{t-1}) \le 2L\sum_{t=1}^T \mathcal{D}_{f_t}(\mathbf{x}^\star, \mathbf{x}_t) + V_T,$$

Technical Challenge



- How to obtain gradient-variation regret?

Two technical routines - II:

$$\bar{V}_T \lesssim \sum_{t=2}^{T} \left(\|\nabla f_t(\mathbf{x}_t) - \nabla f_t(\mathbf{x}^\star)\|^2 + \|\nabla f_t(\mathbf{x}^\star) - \nabla f_{t-1}(\mathbf{x}^\star)\|^2 + \|\nabla f_{t-1}(\mathbf{x}^\star) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|^2 \right)$$

$$\lesssim L \sum_{t=2}^{T} \mathcal{D}_{f_t}(\mathbf{x}^\star, \mathbf{x}_t) + V_T + L \sum_{t=2}^{T} \mathcal{D}_{f_{t-1}}(\mathbf{x}^\star, \mathbf{x}_{t-1}) \leq 2L \sum_{t=1}^{T} \mathcal{D}_{f_t}(\mathbf{x}^\star, \mathbf{x}_t) + V_T,$$

Solution: $$\sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}^\star) = \sum_{t=1}^{T} \langle \nabla f_t(\mathbf{x}_t), \mathbf{x}_t - \mathbf{x}^\star \rangle - \sum_{t=1}^{T} \mathcal{D}_{f_t}(\mathbf{x}^\star, \mathbf{x}_t)$$ (***algorithm-independent!***)

Bregman divergence can be seen as ***compensation from linearization***.

Summary

- Problem: ***universal online learning*** *with* ***gradient variations***

- General framework: ***online ensemble*** ***with adaptivity***

- Applications: ***optimization and online games***

- General analysis: ***meta-base regret decomposition***

- Two approaches: ***correction****-based/****Bregman-div.*** *based* ***cancellation***

Universal online learning with gradient variations: A multi-layer online ensemble approach, NeurIPS'23 **(Spotlight)**
A simple and optimal approach for universal online learning with gradient variations, NeurIPS'24

Thanks!